AWS Systems Manager

Manuel de l'utilisateur

AWS Systems Manager: Manuel de l'utilisateur

Copyright © 2020 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

Présentation du AWS Systems Manager 1
- Capacités 4
  - Configuration rapide 4
  - Gestion des opérations 4
  - Application Management 5
  - Actions et modifications 6
  - Instances et nœuds 6
  - Ressources partagées 8
- Fonctionnement 8
- À propos d'Agent SSM 10
- Systèmes d'exploitation pris en charge 11
  - Linux 11
  - Raspbian 13
  - Windows Server 14
- Accès au Systems Manager 14
- Conditions préalables requises 15

Configuration rapide 17
- Rôles d'autorisations 18
  - Détails sur le profil d'instance par défaut 19
  - Détails sur le rôle de service 20
- Mettre à jour l'agent Systems Manager (SSM) 21
- Collecter l'inventaire à partir de vos instances 21
- Analyser quotidiennement les instances pour rechercher les correctifs manquants 22
- Installer et configurer l'agent CloudWatch 22
- Mettre à jour l'agent CloudWatch toutes les quatre semaines 23
- Choix de cibles pourConfiguration rapide 23
- Utilisation des Configuration rapide résultats 24
- Configuration rapideDépannage des résultats 25
- En cours d'exécution Configuration rapide à nouveau 27

Configuration d'Systems Manager 28
- Étape 1 : Inscrivez-vous à AWS 28
- Étape 2 : Créer un utilisateur IAM administrateur pour AWS 29
- Étape 3 : Créer des utilisateurs et des groupes IAM non administrateurs pour Systems Manager 30
  - Tâche 1 : Créer des groupes d'utilisateurs 31
  - Tâche 2 : Créer des utilisateurs et attribuer des autorisations 32
- Étape 4 : Créer un profil dIAM'instance pourSystems Manager 34
  - À propos de stratégies pour un profil d'instance Systems Manager 34
  - Tâche 1 : (Facultatif) Créer une stratégie personnalisée pour l'accès au compartiment S3 36
  - Tâche 2 : Ajouter des autorisations à un profil dSystems Manager'instance (console) 38
- Étape 5 : Attachement d'un profil d'instance IAM à une instance EC2 40
  - Lancer une instance qui utilise le profil d'instance Systems Manager (console) 41
  - Attacher le profil d'instance Systems Manager à une instance existante (console) 41
- Étape 6 : (Facultatif) Créer un point de terminaison de VPC 42
  - Limites et restrictions applicables aux points de terminaison de VPC 42
  - Création des points de terminaison d'un VPC pour Systems Manager 44
- Étape 7 : (Facultatif) Créer des rôles de service Systems Manager 45
  - Création d'un rôle de service 46
- Étape 8 : (Facultatif) Configuration des intégrations avec d'autres AWS services 47

Configuration des environnements hybrides 49
- Étape 1 : Effectuer les étapes générales Systems Manager de configuration 50
- Étape 2 : Créer un rôle de service IAM pour un environnement hybride 50
- Étape 3 : Installer un certificat TLS sur les serveurs sur site etVMs 54
- Étape 4 : Créer une activation d'instance gérée pour un environnement hybride 55
  - Création d'une activation (console) 56

Création d'une activation d'instance gérée (ligne de commande) 57
Étape 5 : Installer Agent SSM pour un environnement hybride (Linux) 58
Étape 6 : Installer Agent SSM pour un environnement hybride (Windows) 62
Mise en route 65
Étape 1 : Installer ou mettre à niveau les outils de ligne de commande AWS 65
Étape 2 : Exercice pratique d'installation ou de mise à jour Agent SSM sur une instance 67
Étape 3 : Essai des Systems Manager didacticiels et des procédures 68
Utilisation des Agent SSM 72
Agent SSMversion 3 75
Installation et configuration de l'Agent SSM sur les instances EC2 pour Linux 75
Installer manuellement l'Agent SSM sur les instances EC2 pour Linux 76
Configurer Agent SSM pour utiliser un proxy 96
Désinstaller Agent SSM sur les instances Linux 99
Installation et configuration Agent SSM de sur les instances EC2 pourWindows Server 100
Installation manuelle Agent SSM sur les instances EC2 pourWindows Server 100
Configurer l'Agent SSM pour utiliser un proxy pour les instances Windows Server 102
Vérification de Agent SSM l'état et démarrage de l'agent 104
Vérification du numéro de Agent SSM version 106
Affichage des Agent SSM journaux 107
Activation Agent SSM de la journalisation de débogage 107
Restriction de l'accès aux commandes de niveau racine viaAgent SSM 108
Automatisation des mises à jour versAgent SSM 109
Mise à jour automatiqueAgent SSM 110
Abonnement aux notifications Agent SSM 111
À propos des autorisations minimales relatives aux compartiments S3 pour l'Agent SSM 112
Autorisations requises 112
Example 114
Dépannage de Agent SSM 115
nAgent SSM'est pas à jour 115
Afficher les fichiers journaux Agent SSM 115
Intégrations de produits et services 116
Intégration aux AWS services 116
Compute 116
Storage 118
Outils pour développeurs 118
Sécurité, identité et conformité 119
Chiffrement et ICP 120
Gestion et gouvernance 121
Mise en réseau et diffusion de contenu 123
Analytics 124
Intégration d'applications 125
AWS Management Console 125
Exécution de scripts depuis Amazon S3 126
Référencement des secrets AWS Secrets Manager à partir des paramètres Parameter Store 134
Intégration à d'autres produits et services 138
Exécution de scripts depuis GitHub 139
Utilisation des InSpec profils Chef avec Systems Manager la conformité 145
Fonctionnement 146
Exécution d'une analyse InSpec de conformité 146
Exemples d'intégration issus de la communauté 149
Billets de blogs 149
Gestion des opérations 153
Explorer 153
Quelles sont les fonctions d'Explorer ? 154
Quel est le lien entre Explorer et OpsCenter ? 155
Présentation du OpsData 155
L'utilisation d'Explorer entraîne-t-elle des frais ? 156

Mise en route 157
Utilisation de Explorer 168
Exportation de OpsData 172
Dépannage 174
OpsCenter 176
Comment mon organisation peut-elle tirer parti de OpsCenter ? 177
Quelles sont les fonctions d'OpsCenter ? 177
Comment OpsCenter fonctionne-t-il avec Amazon EventBridge ? Quel service dois-je utiliser ? 179
OpsCenter s'intègre-t-il à mon système actuel de gestion des cas ? 179
L'utilisation d'OpsCenter entraîne-t-elle des frais ? 180
Est-ce que OpsCenter fonctionne avec mes instances gérées locales et hybrides ? 180
Quelles sont les limites des ressources pour OpsCenter ? 180
Démarrez avec OpsCenter 181
Création de OpsItems 188
Utilisation des OpsItems 193
Correction des problèmes OpsItem 200
Affichage des rapports de synthèse OpsCenter 206
Référence des ressources prises en charge 206
Audit et journalisation de l'activité de OpsCenter 210
Tableaux de bord CloudWatch 210
Trusted Advisor et PHD 210
Application Management 5
Groupes de ressources 212
Affichage des données d'opérations pour Groupes de ressources AWS 214
AppConfig 218
Parameter Store 218
Comment mon organisation peut-elle tirer parti de Parameter Store ? 219
À qui est destiné Parameter Store ? 219
Quelles sont les fonctions d'Parameter Store ? 219
Qu'est-ce qu'un paramètre ? 220
Configuration d'Parameter Store 223
Utilisation des Parameter Store 242
Procédures Parameter Store 299
Audit et journalisation de l'activité de Parameter Store 302
Dépannage de Parameter Store 302
Actions et modifications 304
Automation 304
Cas d'utilisation Automation 306
Configuration d'Automation 308
Utilisation des automatisations 313
Référence sur les actions Automation 386
Utilisation des documents Automation 446
Référence des documents Automation 511
Procédures d'automatisation 632
Dépannage de Systems Manager Automation 676
Modifier le calendrier 680
À qui est destiné Modifier le calendrier ? 681
Avantages d’Modifier le calendrier 681
Configuration d'Modifier le calendrier 681
Utilisation des Modifier le calendrier 682
Ajouter des dépendances Modifier le calendrier aux documents Automation 688
Fenêtres de maintenance 688
Configuration d'Fenêtres de maintenance 690
Utilisation des fenêtres de maintenance (console) 707
Didacticiels sur les fenêtres de maintenance (AWS CLI) 715
Procédures pas à pas d'une fenêtre de maintenance 763
Options de planification de la fenêtre de maintenance et de période active 772

Fenêtres de maintenance de dépannage 776
Instances et nœuds 778
Conformité 778
Mise en route avec le service Conformité de configuration 779
Création d'une synchronisation de données de ressources pour Conformité de configuration 780
Utilisation de la conformité de configuration 782
Correction des problèmes de conformité à l'aide d'EventBridge 785
Procédure pas à pas de Conformité de la configuration (AWS CLI) 787
Inventaire 791
En savoir plus sur l'inventaire 794
Configuration de la synchronisation de données de ressource pour l'inventaire 800
Configuration de la collecte d'inventaire 806
Utilisation des données d'inventaire 810
Utilisation de l'inventaire personnalisé 826
Affichage de l'historique d'inventaire et suivi des modifications 836
Procédures pas à pas de l'inventaire 837
Résolution des problèmes d'inventaire 849
Instances gérées 851
Configuration des niveaux d'instance 853
Réinitialisation de mots de passe sur des instances gérées 860
Annuler l'enregistrement des instances gérées dans un environnement hybride 864
Dépannage de la disponibilité Amazon EC2 des instances gérées 865
Activations hybrides 868
Session Manager 869
Comment mon organisation peut-elle tirer parti de Session Manager ? 869
À qui est destiné Session Manager ? 870
Quelles sont les principales fonctionnalités Session Manager ? 871
Qu'est-ce qu'une session ? 872
Configuration d'Session Manager 873
Utilisation des Session Manager 917
Audit et journalisation de l'activité de session 929
Schéma du document de session 933
Dépannage de Session Manager 938
Fonctionnalité Exécuter la commande 941
Configuration d'Fonctionnalité Exécuter la commande 943
Exécution des commandes 946
Gestion des codes de sortie avec des scripts 963
Comprendre les états des commandes 965
Procédures Fonctionnalité Exécuter la commande 970
Dépannage de Fonctionnalité Exécuter la commande 986
State Manager 987
À propos d’State Manager 988
Utilisation des associations 989
Procédures State Manager 1019
Gestionnaire de correctifs 1047
Prérequis d'Gestionnaire de correctifs 1048
Fonctionnement 1049
À propos des opérations d'application de correctifs 1076
À propos des références de correctifs 1098
Correction d'instances hors conformité 1110
Utilisation de Kernel Live Patching sur les instances Amazon Linux 2 1117
Utilisation d' Gestionnaire de correctifs (console) 1123
Utilisation dGestionnaire de correctifs' AWS CLI () 1147
Procédures Gestionnaire de correctifs 1166
Distributor 1174
Comment mon organisation peut-elle tirer parti de Distributor ? 1174
À qui est destiné Distributor ? 1175

Quelles sont les fonctions d'Distributor ? ... 1175
Qu'est-ce qu'un package ? ... 1176
Configuration d'Distributor ... 1177
Utilisation des Distributor ... 1179
Audit et journalisation de l'activité de Distributor ... 1208
Dépannage de Distributor ... 1209
Ressources partagées ... 1211
Documents SSM ... 1211
Schémas et fonctionnalités de document SSM ... 1213
Syntaxe du document SSM ... 1226
Référence du plug-in de document Systems Manager ... 1232
Création de documents Systems Manager ... 1267
Partage de documents SSM ... 1275
Recherche de SSM documents ... 1283
Exécution de documents Systems Manager de commande à partir d'emplacements distants ... 1286
Sécurité ... 1290
Protection des données ... 1290
Chiffrement des données ... 1291
Confidentialité du trafic inter-réseaux ... 1293
Identity and Access Management ... 1293
Audience ... 1294
Authentification avec des identités ... 1294
Gestion de l'accès à l'aide de stratégies ... 1296
Fonctionnement d'AWS Systems Manager avec IAM ... 1298
Exemples de stratégies basées sur l'identité ... 1304
Dépannage ... 1310
Utilisation des rôles liés à un service ... 1312
Rôle Windows d'inventaire et de maintenance ... 1312
Rôle de découverte de compte Explorer ... 1315
Journalisation et surveillance ... 1318
Validation de la conformité ... 1320
Résilience ... 1321
Sécurité de l'infrastructure ... 1321
Configuration et analyse des vulnérabilités ... 1321
Bonnes pratiques de sécurité ... 1321
Bonnes pratiques de sécurité préventive pour Systems Manager ... 1322
Bonnes pratiques de surveillance et d'audit pour Systems Manager ... 1324
Surveillance ... 1326
Outils de surveillance ... 1327
Envoi de journaux d'instance à CloudWatch Logs (Agent CloudWatch) ... 1327
Migrer la collecte de journaux d'instances Windows Server vers l'Agent CloudWatch ... 1328
Stocker les paramètres de configuration de l'Agent CloudWatch dans Parameter Store ... 1332
Retourner à la collecte de journaux avec l'Agent SSM ... 1332
Envoi de Agent SSM journaux àCloudWatch Logs ... 1333
Surveillance des métriques Fonctionnalité Exécuter la commande avec Amazon CloudWatch ... 1335
Dimensions et métriques Systems Manager Fonctionnalité Exécuter la commande ... 1336
Journalisation des appels d'API AWS Systems Manager avec AWS CloudTrail ... 1336
Informations Systems Manager dans CloudTrail ... 1337
Présentation des entrées des fichiers journaux Systems Manager ... 1337
Configuration d'Amazon CloudWatch Logs pour Fonctionnalité Exécuter la commande ... 1338
Spécification de CloudWatch Logs lorsque vous envoyez des commandes ... 1339
Affichage de la sortie de commande dans CloudWatch Logs ... 1340
Surveillance avec Amazon EventBridge ... 1340
Configuration de EventBridge pour Systems Manager événements ... 1341
Systems Manager exemples de type de cible ... 1343
Surveillance des changements de statut Systems Manager à l'aide de notifications Amazon SNS ... 1344
Configurer les notifications Amazon SNS pour AWS Systems Manager ... 1345

Exemples de notifications Amazon SNS pour AWS Systems Manager 1351
Utiliser la fonctionnalité Fonctionnalité Exécuter la commande pour envoyer une commande qui renvoie des notifications de statut 1352
Utilisation d'une fenêtre de maintenance pour envoyer une commande qui renvoie des notifications de statut 1354
Balisage des ressources Systems Manager 1359
Ressources Systems Manager pouvant être balisées 1359
Balisage de documents Systems Manager 1360
Création de documents avec des balises 1360
Ajout de balises à des documents existants 1361
Suppression de balises des documents Systems Manager 1363
Balisage des fenêtres de maintenance 1365
Création de fenêtres de maintenance avec des balises 1365
Ajout de balises aux fenêtres de maintenance existantes 1365
Suppression de balises des fenêtres de maintenance 1367
Balisage des instances gérées 1369
Création ou activation d'instances gérées avec des balises 1369
Ajout de balises à des instances gérées existantes 1370
Suppression de balises des instances gérées 1372
BalisageOpsItems 1374
Création avec des balisesOpsItems 1374
Ajout de balises à desOpsItems 1374
Suppression de balises d'Systems ManagerOpsItems 1376
Balisage des paramètres Systems Manager 1377
Création de paramètres avec des balises 1378
Ajout de balises aux paramètres existants 1378
Suppression de balises des paramètres Systems Manager 1380
Balisage des références de correctifs 1381
Création de références de correctifs avec des balises 1382
Ajout de balises aux références de correctifs existantes 1382
Suppression de balises des références de correctifs 1384
Référence AWS Systems Manager 1387
Modèles et types EventBridge d'événements pourSystems Manager 1387
Type d'événement : Automation 1388
Type d'événement : Conformité de la configuration 1389
Type d'événement : Inventaire 1389
Type d'événement :State Manager 1389
Type d'événement : Fenêtre de maintenance 1390
Type d'événement :Parameter Store 1391
Type d'événement :Fonctionnalité Exécuter la commande 1392
Expressions cron et rate 1392
Informations générales sur les expressions cron et rate 1394
Expressions cron et rate pour les associations 1397
Expressions cron et rate pour les fenêtres de maintenance 1398
ec2messages, ssmmessages et autres appels d'API 1399
Création de chaînes de date et d'heure formatées pour Systems Manager 1400
Mise en forme de chaînes de date et d'heure pour Systems Manager 1400
Création de chaînes de date et d'heure personnalisées pour Systems Manager 1401
Cas d'utilisation et bonnes pratiques 1403
Choisir entre State Manager etFenêtres de maintenance 1405
State Manageret Fenêtres de maintenance : Cas d'utilisation des clés 1405
Historique du document 1409
Mises à jour antérieures 1475
Glossaire AWS 1489
..... mcdxc

Présentation du AWS Systems Manager

AWS Systems Manager est un service AWS que vous pouvez utiliser pour afficher et contrôler votre infrastructure sur AWS. À l'aide de la console Systems Manager, vous pouvez afficher les données opérationnelles de plusieurs services AWS et automatiser des tâches opérationnelles dans vos différentes ressources AWS. Systems Manager vous aide à gérer la sécurité et la conformité en analysant vos instances gérées, et en signalant les violations de stratégie détectées (ou en prenant des mesures correctives sur celles-ci).

Une instance gérée est une machine qui a été configurée pour une utilisation avec Systems Manager. Systems Manager vous permet également de configurer et gérer vos instances gérées. Les types de machine pris en charge incluent les instances EC2, les serveurs sur site et les machines virtuelles, y compris VMs dans d'autres environnements cloud. Les types de système d'exploitation pris en charge incluent Windows Server, plusieurs distributions de Linux, et Raspbian.

Avec Systems Manager, vous pouvez associer des ressources AWS en appliquant la même balise de ressource d'identification à chacune d'entre elles. Vous pouvez ensuite afficher les données opérationnelles pour ces ressources en tant que groupe de ressources, pour vous aider à les surveiller et les dépanner.

Par exemple, vous pouvez attribuer une balise de ressource « `Operation=North Region OS Patching` » à toutes les ressources suivantes :

- Groupe d'instances EC2
- Un groupe de serveurs sur site dans vos propres installations
- Une référence de correctifs Systems Manager qui spécifie les correctifs à appliquer à vos instances gérées
- Compartiment S3 pour stocker la sortie de journal de l'opération d'application de correctifs
- Une fenêtre de maintenance Systems Manager qui spécifie le calendrier pour l'opération d'application de correctifs

Après avoir balisé les ressources, vous pouvez afficher un tableau de bord consolidé dans Systems Manager qui indique le statut de toutes les ressources qui font partie de l'opération d'application de correctifs dans votre région North. Si un problème survient avec l'un de ces ressources, vous pouvez immédiatement prendre des mesures correctives.

Fonctionnalités de Systems Manager

Systems Managerest composé de capacités individuelles (p. 4) qui sont regroupées en cinq catégories : Gestion des opérations, gestion des applications, actions et modifications, instances et nœuds et ressources partagées.

Cette collection de fonctionnalités constitue un ensemble de fonctions et d'outils puissants que vous pouvez utiliser pour effectuer de nombreuses tâches opérationnelles. Exemple :

- Regrouper des ressources AWS selon tout objectif ou activité que vous choisissez, par exemple, application, environnement, région, projet, campagne, unité commerciale, ou cycle de vie des logiciels.
- Définir de façon centralisée les options de configuration et les stratégies pour vos instances gérées.

- Afficher, investiguer et résoudre de façon centralisée les éléments de travail opérationnels liés aux ressources AWS.
- Automatiser ou planifier une grande variété de tâches de maintenance et de déploiement.
- Utiliser et créer des documents SSM de style de runbook qui définissent les actions à exécuter sur vos instances gérées.
- Exécuter une commande, avec contrôles de taux et d'erreur, qui cible l'ensemble d'un parc d'instances gérées.
- Vous connecter en toute sécurité à une instance gérée d'un simple clic, sans avoir à ouvrir un port entrant ou à gérer des clés SSH.
- Séparer vos secrets et vos données de configuration de votre code à l'aide de paramètres, avec ou sans chiffrement, puis faire référence à ces paramètres depuis différents autres services AWS.
- Effectuer un inventaire automatisé en collectant des métadonnées sur vos instances gérées Amazon EC2 et sur site. Les métadonnées peuvent inclure des informations sur les applications, les configurations réseau, etc.
- Afficher les données d'inventaire consolidées provenant de plusieurs régions AWS et comptes que vous gérez.
- Voir rapidement quelles ressources de votre compte sont non conformes et prendre des mesures correctives à partir d'un tableau de bord centralisé.
- Afficher des récapitulatifs actifs de métriques et d'alarmes pour vos ressources AWS.

Systems Manager simplifie la gestion des ressources et des applications, raccourcit le délai de détection et de résolution des problèmes opérationnels, et vous aide à exploiter et gérer votre infrastructure AWS de façon sécurisée à grande échelle.

> **Note**
>
> AWS Systems Managerétait auparavant connu sous le nom d'Amazon Simple Systems Manager (SSM) et Amazon EC2 Systems Manager (SSM). Pour plus d'informations, consultez Systems Manager service name history (p. 2) .

Présentation de AWS Systems Manager (Vidéo)

Voir plus de vidéos AWS sur la Amazon Web Services YouTube chaîne .

Systems ManagerRégionAWSs prises en charge par

AWS Systems Manager est disponible dans les régions AWS répertoriées dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services. Avant de démarrer votre processus de configuration de Systems Manager, nous vous recommandons de vous assurer que le service est disponible dans chacune des régions AWS que vous souhaitez utiliser.

Pour les serveurs sur site et VMs dans votre environnement hybride, nous vous recommandons de choisir la région la plus proche de votre centre de données ou de votre environnement informatique.

Tarification de Systems Manager

Certaines fonctionnalités Systems Manager font l'objet d'une facturation. Pour de plus amples informations, veuillez consulter Tarification AWS Systems Manager.

historique des noms de Systems Manager service

AWS Systems Manager(Systems Manager) était auparavant connu sous le nom de «Amazon Simple Systems Manager (SSM)" and "Amazon EC2 Systems Manager (SSM)". Le nom abrégé d'origine du service, «SSMest toujours reflété dans diverses AWS ressources, y compris dans d'autres consoles de service. Voici quelques exemples :

- Systems Manager Agent: Agent SSM

- Systems ManagerParamètres : Paramètres SSM
- Points de terminaison du service Systems Manager : `ssm.us-east-2.amazonaws.com`
- Types de ressource AWS CloudFormation : `AWS::SSM::Document`
- Identifiant de la règle AWS Config : `EC2_INSTANCE_MANAGED_BY_SSM`
- Commandes de l'AWS CLI : `aws ssm describe-patch-baselines`
- Noms de la stratégie gérée AWS Identity and Access Management (IAM) : `AmazonSSMReadOnlyAccess`
- Systems Managerressource ARNs :`arn:aws:ssm:us-east-2:111222333444:patchbaseline/pb-07d8884178EXAMPLE`

Contenu connexe

- Les ressources suivantes peuvent vous aider à utiliser Systems Manager.
  - Blog & Podcast AWS – Lisez des billets de blog sur Systems Manager dans la catagorie AWS Management Tools, ainsi que les autres billets avec la balise `#Systems Manager`.
  - Forum de développeurs Systems Manager – Suivez les annonces, ou publiez ou répondez à une question dans le forum AWS Systems Manager.
  - Référence d'API AWS Systems Manager – Fournit des descriptions, la syntaxe et des exemples d'utilisation pour chacun des types de données et des actions Systems Manager.
  - AWS Systems Manager section of the AWS CLI Command Reference – Gérez Systems Manager à partir d'un outil de ligne de commande. Disponible sur les systèmes Windows, Mac et Linux/UNIX.
  - AWS Systems Manager section of the Outils AWS pour PowerShell Cmdlet Reference–Gérez Systems Manager avec les mêmes PowerShell outils que ceux que vous utilisez pour gérer vos environnements Windows, Linux ou Mac.
  - Quotas de service Systems Manager dans le Référence générale d'Amazon Web Services. – Fournit les quotas par défaut pour Systems Manager pour un compte AWS. Sauf indication contraire, chaque quota est spécifique à la région.

  Les ressources connexes suivantes peuvent s'avérer utiles lors de l'utilisation de ce service.
  - Formations et ateliers – Liens vers des formations spécialisées et basées sur les rôles, ainsi que des ateliers d'autoformation pour améliorer vos compétences AWS et acquérir une expérience pratique.
  - Outils de développement AWS– Liens vers des outils de développement, kits SDK, boîtes à outils IDE et outils de ligne de commande pour développer et gérer des applications AWS.
  - Livres blancs AWS – Liens vers une liste complète des livres blancs techniques AWS, couvrant des sujets tels que l'architecture, la sécurité et l'économie, créés par des architectes de solutions AWS ou d'autres experts techniques.
  - Centre AWS Support – Le hub pour la création et la gestion de vos demandes AWS Support. Inclut également des liens vers d'autres ressources utiles, telles que des forums, des FAQ techniques, l'état d'un service et AWS Trusted Advisor.
  - AWS Support – Page principale Web d'informations à propos d'AWS Support, un canal d'assistance technique individuelle rapide pour vous aider à développer et à exécuter des applications dans le Cloud.
  - Contactez-nous – Point de contact central pour toute question relative à la facturation AWS, à votre compte, aux événements, à des abus ou à d'autres problèmes.
  - Conditions d'utilisation du site AWS – Informations détaillées sur nos droits d'auteur et notre marque, sur vos compte, licence et accès au site, et sur d'autres sujets.

En savoir plus sur Systems Manager

- Capacités Systems Manager (p. 4)
- Fonctionnement de Systems Manager (p. 8)

- À propos d'Agent SSM (p. 10)
- Systèmes d'exploitation pris en charge (p. 11)
- Accès au Systems Manager (p. 14)
- Prérequis d'Systems Manager (p. 15)

Capacités Systems Manager

Les fonctionnalités de Systems Manager sont regroupées selon les types de fonctionnalités suivants :

Rubriques

- Configuration rapide (p. 4)
- Gestion des opérations (p. 4)
- Application Management (p. 5)
- Actions et modifications (p. 6)
- Instances et nœuds (p. 6)
- Ressources partagées (p. 8)

Configuration rapide

La Configuration rapide (p. 17) est un outil qui permet de configurer rapidement les rôles de sécurité requis et les fonctionnalités Systems Manager couramment utilisées sur vos instances EC2. Ces fonctionnalités vous aident à gérer et à surveiller l'état de vos instances tout en fournissant les autorisations minimales requises pour démarrer. Plus précisément, la configuration rapide vous aide à configurer les composants suivants sur les instances que vous choisissez ou ciblez à l'aide de balises :

- Rôles de profil d'instanceAWS Identity and Access Management (IAM) pour Systems Manager.
- Une mise à jour planifiée bihebdomadaire de Agent SSM.
- Une collecte planifiée de métadonnées d'inventaire toutes les 30 minutes.
- Une analyse quotidienne de vos instances pour identifier les correctifs manquants.
- Une installation et une configuration uniques de l'agent Amazon CloudWatch.
- Une mise à jour mensuelle planifiée de l'agent CloudWatch.

Gestion des opérations

La gestion des opérations est une suite de fonctionnalités qui vous aident à gérer vos AWS ressources.

Explorer

Explorer (p. 153)est un tableau de bord des opérations personnalisable qui contient des informations sur vos AWS ressources. Explorer affiche une vue agrégée des données d'opérations (OpsData) pour vos AWS comptes et dans toutes les régions. Dans Explorer , OpsData inclut des métadonnées sur vos instances EC2, des détails de conformité des correctifs et des éléments de travail opérationnels (OpsItems). Explorer fournit un contexte sur la répartition entre vos unités commerciales ou applications, leur évolution au fil du temps et leur variation par catégorie.OpsItems Vous pouvez regrouper et filtrer les informations dans Explorer pour vous concentrer sur les éléments qui vous intéressent et qui nécessitent une action. Lorsque vous identifiez des problèmes prioritaires, vous pouvez utiliser Systems Manager OpsCenter pour exécuter des runbooks Automation et résoudre rapidement ces problèmes.

OpsCenter

OpsCenter (p. 176)fournit un emplacement central où les ingénieurs d'exploitation et IT Les professionnels peuvent afficher, étudier et résoudre les éléments de travail opérationnels (OpsItems) liés aux ressources AWS. OpsCenter est conçu pour réduire le temps moyen de résolution des problèmes ayant un impact sur les ressources AWS. Cette fonctionnalité Systems Manager regroupe et normalise les OpsItems entre les services tout en fournissant des données d'investigation contextuelles sur chaque OpsItem, sur les OpsItems associés et sur les ressources connexes. OpsCenter fournit également des documents Automation Systems Manager (runbooks) que vous pouvez utiliser pour résoudre rapidement les problèmes. Vous pouvez spécifier des données personnalisées consultables pour chaque OpsItem. Vous pouvez également afficher des rapports récapitulatifs sur les OpsItems générés automatiquement, par statut et par source.

CloudWatch Dashboards

Les tableaux de bord Amazon CloudWatch sont des pages d'accueil personnalisables de la console CloudWatch que vous pouvez utiliser pour surveiller vos ressources dans une seule vue, y compris les ressources réparties sur différentes régions. Vous pouvez utiliser les tableaux de bord CloudWatch pour créer des vues personnalisées des métriques et alarmes de vos ressources AWS.

Trusted Advisor & Tableau de bord d'état personnel (PHD)

Systems Managerhéberge deux outils en ligne pour vous aider à mettre en service vos ressources et surveiller votre compte pour les événements d'intégrité. Trusted Advisor est un outil en ligne qui vous fournit des conseils en temps réel pour vous aider à mettre en service vos ressources conformément aux AWS bonnes pratiques. Pour plus d'informations, consultez la section Trusted Advisor.

Le AWS Personal Health Dashboard fournit des informations sur les événements AWS d'état qui peuvent avoir un impact sur votre compte . Les informations sont présentées de deux manières : un tableau de bord qui montre les événements récents et à venir organisés par catégorie, et un journal des événements complet qui contient tous les événements des 90 derniers jours. Pour plus d'informations, consultez Démarrer avec le AWS Personal Health Dashboard.

Application Management

Application Management est une suite de fonctionnalités qui vous aident à gérer vos applications exécutées dans AWS .

Resource Groups

Groupes de ressources AWS : Une AWS ressource est une entité que vous pouvez utiliser dans AWS , comme les documents Systems Manager SSM, les références de correctifs, les fenêtres de maintenance, les paramètres et les instances gérées, une instance Amazon Elastic Compute Cloud (EC2), un volume Amazon Elastic Block Store (Amazon EBS), un groupe de sécurité ou un Amazon Virtual Private Cloud (VPC. Un groupe de ressources est un ensemble de AWS ressources qui se trouvent toutes dans la même AWS région et qui correspondent aux critères indiqués dans une requête. Créez des requêtes dans la console Groupes de ressources ou transmettez-les sous forme d'arguments aux Resource Groups commandes dans l' AWS CLI . Avec Resource Groups, vous pouvez créer une console personnalisée qui organise et consolide les informations dont vous avez besoin en fonction des critères indiqués dans des balises. Vous pouvez également utiliser des groupes comme base pour afficher les analyses de surveillance et de configuration dans AWS Systems Manager.

AWS AppConfig

AppConfig (p. 218) permet de créer, gérer et déployer rapidement des configurations d'applications. AppConfig prend en charge les déploiements contrôlés vers des applications de toute taille. Vous pouvez utiliser AppConfig avec des applications hébergées sur des instances EC2, AWS Lambda , des conteneurs, des applications mobiles ou des IoT appareils. Pour éviter les erreurs lors du déploiement

de configurations d'application, AppConfig propose des validateurs. Un validateur permet une vérification syntaxique ou sémantique pour s'assurer que la configuration que vous souhaitez déployer fonctionne comme prévu. Lors d'un déploiement de configuration, AppConfig surveille l'application pour s'assurer que le déploiement a réussi. Si le système rencontre une erreur ou si le déploiement déclenche une alarme, AppConfig annule la modification afin de minimiser l'impact sur les utilisateurs de votre application.

Parameter Store

Parameter Store (p. 218) offre un stockage sécurisé et hiérarchique des données de configuration et de gestion des codes secrets. Vous pouvez stocker des données telles que des mots de passe, des chaînes de base de données, des instances EC2 IDs et des Amazon Machine Image (AMI) et les codes de licence comme valeurs des paramètres.IDs Vous pouvez stocker ces valeurs sous forme de texte brut ou de données chiffrées. Vous pouvez ensuite faire référence à ces valeurs en utilisant le nom unique que vous avez spécifié lors de la création du paramètre.

Actions et modifications

Systems Manager fournit les fonctions suivantes pour exécuter des actions sur vos ressources AWS ou les modifier.

Automation

Utilisez la fonction d'automatisation Systems Manager Automation (p. 304) pour automatiser les tâches courantes de maintenance et de déploiement. Vous pouvez utiliser Automation pour créer et mettre à jour Amazon Machine Images, appliquer les mises à jour du pilote et de l'agent, réinitialiser les mots de passe sur Windows Server l'instance, réinitialiser les clés SSH sur les instances Linux, et appliquer OS Correctifs ou mises à jour de l'application.

Modifier le calendrier

Modifier le calendrier (p. 680) vous permet de configurer des plages de dates et de temps pendant lesquelles les actions que vous spécifiez (par exemple, dans des documents Systems Manager Automation (p. 304)) peuvent ou non être effectuées dans votre compte AWS. Dans Modifier le calendrier, ces plages sont appelées des événements. Lorsque vous créez une Modifier le calendrier entrée, vous créez un Systems Manager document (p. 1211) de type `ChangeCalendar` . Dans Modifier le calendrier , le document stocke les données iCalendar 2.0 au format texte brut. Les événements que vous ajoutez à l'entrée Modifier le calendrier font partie du document.

Fenêtres de maintenance

Utilisez les Fenêtres de maintenance (p. 688) pour configurer des planifications récurrentes pour les instances gérées afin d'exécuter des tâches d'administration telles que l'installation de correctifs et de mises à jour sans interrompre les opérations stratégiques.

Instances et nœuds

Systems Managerfournit des fonctionnalités pour gérer vos instances EC2, vos serveurs sur site et les machines virtuelles de votre environnement hybride, ainsi que d'autres types de AWS ressources (nœuds).

Compliance

Utilisez le service Conformité de configuration Systems Manager (p. 778) pour analyser votre parc d'instances gérées afin de rechercher des incohérences de conformité et de configuration de correctifs. Vous pouvez collecter et regrouper des données de plusieurs comptes et régions AWS, puis explorer plus en détails des ressources spécifiques qui ne sont pas conformes. Par défaut, le service Conformité de configuration affiche les données de conformité relatives Gestionnaire de correctifs à

l'application des correctifs et State Manager aux associations. Vous pouvez également personnaliser le service et créer vos propres types de conformité en fonction de vos IT ou exigences métier.

Inventory

L'inventaire (p. 791) automatise le processus de collecte de l'inventaire logiciel à partir d'instances gérées. Vous pouvez utiliser Inventory pour collecter des métadonnées sur les applications, les fichiers, les composants, les correctifs, etc. de vos instances gérées.

Managed Instances

Une instance gérée (p. 28) est une instance EC2 ou une machine sur site – un serveur ou une machine virtuelle – de votre environnement hybride, configurée pour Systems Manager. Pour configurer des instances gérées, vous devez installer l'Agent SSM sur vos machines (s'il n'est pas installé par défaut) et configurer des autorisations AWS Identity and Access Management (IAM). Les machines sur site requièrent également un code d'activation.

Hybrid Activations

Pour configurer des serveurs et VMs dans votre environnement hybride en tant qu'instances gérées, vous devez créer une activation d'instance gérée (p. 49). Une fois que vous avez terminé l'activation, vous recevez un identifiant et un code d'activation. Cette combinaison code/ID fonctionne comme un ID d'accès et une clé secrète Amazon EC2. Elle sécurise l'accès au service Systems Manager à partir de vos instances gérées.

Session Manager

Utilisez Session Manager (p. 869) pour gérer vos instances EC2 via un shell interactif en un clic, basé sur un navigateur, ou via l'AWS CLI. Session Manager vous assure une gestion des instances sécurisée et vérifiable, sans nécessité d'ouvrir les ports entrants ni de gérer des hôtes bastion ou des clés SSH. Session Manager facilite également la mise en conformité avec des stratégies d'entreprise qui nécessitent un accès contrôlé aux instances, des pratiques de sécurité strictes, ainsi que des journaux totalement vérifiables contenant des informations d'accès aux instances, tout en fournissant aux utilisateurs finaux un accès inter-plateforme à vos instances EC2 d'un simple clic.

Fonctionnalité Exécuter la commande

Utilisez Systems Manager Fonctionnalité Exécuter la commande (p. 941) pour gérer à distance et en toute sécurité la configuration de vos instances gérées à grande échelle. Permet Fonctionnalité Exécuter la commande d'effectuer des modifications à la demande telles que la mise à jour d'applications ou l'exécution de scripts shell Linux et Windows PowerShell sur un ensemble cible de dizaines ou de centaines d'instances.

State Manager

Utilisez Systems Manager State Manager (p. 987) pour automatiser le processus permettant de conserver vos instances gérées à un état défini. Vous pouvez utiliser State Manager pour vous assurer que vos instances sont amorcées avec un logiciel spécifique au démarrage, jointes à un Windows Domaine (Windows Serverinstances uniquement) ou corrigé avec des mises à jour logicielles spécifiques.

Gestionnaire de correctifs

Utilisez Gestionnaire de correctifs (p. 1047) pour automatiser le processus d'application des correctifs sur vos instances gérées avec les mises à jour liées à la sécurité et autres. Vous pouvez utiliser Gestionnaire de correctifs pour appliquer des correctifs pour les systèmes d'exploitation et les applications. (Sur Windows Server , la prise en charge des applications est limitée aux mises à jour des applications Microsoft.) Cette fonctionnalité vous permet de rechercher s'il manque des correctifs dans les instances et d'appliquer les correctifs manquants individuellement ou à de grands groupes d'instances en utilisant des balises d'instance EC2. Gestionnaire de correctifs utilise les références de correctifs, qui peuvent inclure des règles pour approuver automatiquement les correctifs quelques jours après leur publication, ainsi qu'une liste des correctifs approuvés et rejetés. Vous pouvez installer des correctifs de sécurité régulièrement en planifiant leur exécution sous forme Systems Manager de

tâche de fenêtre de maintenance, ou vous pouvez appliquer des correctifs à vos instances gérées à la demande à tout moment. Pour les systèmes d'exploitation Linux, vous pouvez définir les référentiels qui doivent être utilisés pour les opérations d'application de correctifs dans le cadre de votre référence de correctifs. Cela vous permet de vous assurer que les mises à jour sont installées uniquement à partir de référentiels approuvés indépendamment des référentiels configurés sur l'instance. Pour Linux, vous avez également la possibilité de mettre à jour n'importe quel package sur l'instance, pas seulement ceux qui sont classés en tant que mises à jour de sécurité du système d'exploitation. Pour Windows Server , vous pouvez également utiliser Gestionnaire de correctifs pour mettre à jour les applications Microsoft prises en charge.

Distributor

permet Distributor (p. 1174) de créer et de déployer des packages sur des instances gérées. vous Distributor permet d'empaqueter vos propres logiciels ou de trouver les packages logiciels d'agent fournis par AWS , tels que AmazonCloudWatchAgent , pour les installer sur les instances AWS Systems Manager gérées. Après avoir installé un package pour la première fois, vous pouvez utiliser Distributor pour désinstaller complètement et réinstaller une nouvelle version du package, ou effectuer une mise à jour sur place qui ajoute uniquement des fichiers nouveaux ou modifiés. Distributor publie des ressources, telles que des packages logiciels, sur des instances AWS Systems Manager gérées.

Ressources partagées

Systems Manager utilise les ressources partagées suivantes pour gérer et configurer vos ressources AWS.

Documents

Un document Systems Manager (p. 1211) (document SSM) définit les actions effectuées par Systems Manager. Les types de document SSM incluent les documents de commande qui sont utilisés par State Manager et Fonctionnalité Exécuter la commande, et les documents Automation, qui sont utilisés par Systems Manager Automation. Systems Manager comporte plusieurs dizaines de documents préconfigurés que vous pouvez utiliser en spécifiant des paramètres lors de l'exécution. Les documents peuvent être exprimés au format JSON ou YAML et incluent les étapes et paramètres que vous spécifiez.

Fonctionnement de Systems Manager

Le diagramme 1 ci-dessous présente un exemple général des différents processus exécutés par Systems Manager lors de l'exécution d'une action, telle qu'envoyer une commande à votre flotte de serveurs ou exécuter un inventaire des applications en cours d'exécution sur vos serveurs sur site. Chaque fonctionnalité Systems Manager, par exemple Fonctionnalité Exécuter la commande et Fenêtres de maintenance, utilise un processus similaire de configuration, d'exécution, de traitement et de création de rapports.

1. – Configurer Systems Manager : Utilisez la Systems Manager console , le kit SDK AWS CLI , l' ou Outils AWS pour Windows PowerShell l' pour configurer, planifier, automatiser et exécuter les actions que vous souhaitez exécuter sur vos AWS ressources .
2. Vérification et traitement : Systems Manager vérifie les configurations, y compris les autorisations, et envoie des demandes à l'Agent SSM en cours d'exécution sur vos instances ou serveurs dans votre environnement hybride. L'Agent SSM effectue les modifications de configuration spécifiées.
3. Génération de rapports : l'Agent SSM indique l'état des modifications et actions de configuration à Systems Manager dans le cloud AWS. Systems Manager envoie ensuite le statut à l'utilisateur et aux divers services AWS, si cette action est configurée.

Diagramme 1 : Exemple général de flux de Systems Manager processus

AWS SDKs
AWS Toolkit for Windows PowerShell
AWS Systems Manager Console
AWS CLI
1
AWS Cloud
Built-in insights
3
VPC
2
Amazon EC2 Windows instances
AWS CloudTrail
AWS Systems Manager
Amazon S3
VPC
Amazon CloudWatch
Amazon EC2 Linux instances
AWS Region 1
2
Linux servers and VMs
Windows servers and VMs
On-premises hybrid environment

À propos d'Agent SSM

AWS Systems Manager Agent (Agent SSM) est un logiciel Amazon qui peut être installé et configuré sur une instance EC2, un serveur sur site ou une machine virtuelle (VM). L'Agent SSM permet à Systems Manager de mettre à jour, de gérer et de configurer ces ressources. L'agent traite les demandes du service Systems Manager dans le cloud AWS et les exécute ensuite comme spécifié dans la demande. Puis, Agent SSM renvoie les informations relatives à l'état et à l'exécution au service Systems Manager à l'aide de Amazon Message Delivery Service (préfixe du service : `ec2messages`).

Agent SSMdoit être installé sur chaque instance que vous souhaitez utiliser avec Systems Manager . Agent SSM est préinstallé, par défaut, sur les instances créées à partir des éléments suivants Amazon Machine Images (AMIs):

- Amazon Linux
- Amazon Linux 2
- AMIs optimisées pour ECS Amazon Linux 2
- Ubuntu Server 16.04, 18.04 et 20.04
- AMIs Windows Server 2008-2012 R2 publiées à partir de novembre 2016
- Windows Server 2016 et 2019

> **Note**
>
> L'Agent SSM n'est pas installé sur toutes les AMIs basées sur Amazon Linux ou Amazon Linux 2. Par exemple, l'Agent SSM n'est pas préinstallé sur les AMIs optimisées pour EKS basées sur Amazon Linux 2.

Sur un autre AMIs, et sur des serveurs et des machines virtuelles sur site pour votre environnement hybride, vous devez installer l'agent manuellement, comme décrit dans le tableau ci-dessous.

> **Important**
>
> Une version mise à jour de l'Agent SSM est publiée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Si une version antérieure de l'agent est en cours d'exécution sur une instance, certains processus de l'Agent SSM peuvent échouer. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir l'Agent SSM à jour sur vos instances. Pour de plus amples informations, veuillez consulter Automatisation des mises à jour versAgent SSM (p. 109). Pour être alerté des mises à jour de Agent SSM, abonnez-vous à la page Agent SSM Release Notes sur GitHub.

Type de système d'exploitation	Installation Agent SSM
Windows	Windows AMIs publiée avant novembre 2016 utilisez le EC2Config Service pour traiter les demandes et configurer les instances. À moins que vous n'ayez une raison spécifique d'utiliser l' EC2Config pour traiter Agent SSM les demandes, nous vous recommandons de télécharger et d'installer la dernière version de l' Systems Manager sur chacune de vos instances EC2 et instances gérées dans votre environnement hybride.Agent SSM Pour plus d'informations, consultez la section Installation et configuration

Type de système d'exploitation	Installation Agent SSM
	Agent SSM de sur les instances EC2 pourWindows Server (p. 100).
Linux	Agent SSMest installé par défaut sur Amazon Linux , Amazon Linux 2 , Ubuntu Server 16.04 et Ubuntu Server 18.04 LTS de base EC2.AMIs Vous devez installer manuellement Agent SSM sur les autres versions d' Amazon EC2 pour Linux, y compris les images ne faisant pas partie de la base de données. Pour plus d'informations, consultez la section Installation et configuration de l'Agent SSM sur les instances EC2 pour Linux (p. 75).
Serveurs sur site et VMs	L'Agent SSM doit être installé manuellement sur les serveurs sur site et les machines virtuelles (VM) que vous souhaitez utiliser dans un environnement hybride. Le processus de téléchargement et d'installation de l'Agent SSM pour ces machines est différent de celui utilisé pour les instances EC2. Pour de plus amples informations, consultez les rubriques suivantes : • Installer Agent SSM pour un environnement hybride (Windows) (p. 62) • Installer Agent SSM pour un environnement hybride (Linux) (p. 58)

Systèmes d'exploitation pris en charge

Vos instances EC2, serveurs sur site et machines virtuelles doivent exécuter l'un des systèmes d'exploitation suivants pour pouvoir être utilisés avec AWS Systems Manager.

Types de système d'exploitation

- Linux (p. 11)
- Raspbian (p. 13)
- Windows Server (p. 14)

Linux

Amazon Linux

Versions	Intel 32-bit (x86)	Intel 64-bit (x86_64)	ARM 64-bit (arm64)
2012.03 – 2018.03	✓	✓	

> **Note**
>
> Depuis la version 2015.03, Amazon Linux est disponible uniquement dans Intel 64-bit (x86_64) les versions .

Amazon Linux 2

Versions	Intel 32-bit (x86)	Intel 64-bit (x86_64)	ARM 64-bit (arm64)
2.0 et toutes les versions ultérieures		✓	✓

CentOS

Versions	Intel 32-bit (x86)	Intel 64-bit (x86_64)	ARM 64-bit (arm64)
6.0	✓		
6.3 et versions 6.x ultérieures	✓	✓	
7.1 et versions 7.x ultérieures		✓	✓
Versions 8.0, 8.1 et 8.2		✓	

Debian Server

Versions	Intel 32-bit (x86)	Intel 64-bit (x86_64)	ARM 64-bit (arm64)
Jessie : (8)		✓	
Stretch (9)		✓	
Buster (10)		✓	

Oracle Linux

Versions	Intel 32-bit (x86)	Intel 64-bit (x86_64)	ARM 64-bit (arm64)
7.5		✓	
7.6		✓	
7.7		✓	
7.8		✓	
7,9		✓	

Red Hat Enterprise Linux (RHEL)

Versions	Intel 32-bit (x86)	Intel 64-bit (x86_64)	ARM 64-bit (arm64)
6.0		✓	
6,5	✓	✓	
6.9	✓	✓	
6.10		✓	

Versions	Intel 32-bit (x86)	Intel 64-bit (x86_64)	ARM 64-bit (arm64)
7.0		✓	
7.1		✓	
7.2		✓	
7.3		✓	
7.4		✓	
7.5		✓	
7.6		✓	✓
7.7		✓	✓
7.8		✓	✓
8.0		✓	✓
8.1		✓	✓
8.2		✓	✓

SUSE Linux Enterprise Server (SLES)

Versions	Intel 32-bit (x86)	Intel 64-bit (x86_64)	ARM 64-bit (arm64)
12 et versions 12.x ultérieures		✓	
15 et versions 15.x ultérieures		✓	

Ubuntu Server

Versions	Intel 32-bit (x86)	Intel 64-bit (x86_64)	ARM 64-bit (arm64)
12.04 LTS et 14.04 LTS	✓	✓	
16.04 LTS et 18.04 LTS		✓	✓
STR 20.04 LTS et 20.10		✓	✓

Raspbian

Version	ARM 32-bit (arm)
Jessie	✓
Stretch	✓

Windows Server

Version	Intel 32-bit (x86)	Intel 64-bit (x86_64)	ARM 64-bit (arm64)
2008	✓	✓	
2008 R2		✓	
2012 et 2012 R2		✓	
2016		✓	
2019		✓	

> **Important**
>
> Agent SSMnécessite Windows PowerShell 3.0 ou version ultérieure pour exécuter certains SSM documents sur Windows Server les instances (par exemple, le `AWS-ApplyPatchBaseline` document hérité). Vérifier que vos Windows Server instances sont en cours d'exécution Windows Management Framework 3.0 ou version ultérieure. Cette infrastructure inclut Windows PowerShellPour de plus amples informations, veuillez consulter . Windows Management Framework 3.0.

Accès au Systems Manager

Vous pouvez utiliser Systems Manager de l'une des façons suivantes :

Console Systems Manager

La console AWS Systems Manager est une interface basée sur un navigateur qui permet d'accéder à Systems Manager et de l'utiliser.

Outils de ligne de commande AWS

Les outils de ligne de AWS commande vous permettent d'envoyer des commandes à la ligne de commande de votre système pour effectuer des Systems Manager tâches AWS et autres, et est pris en charge sur Windows, Linux et macOS . L'utilisation de l' AWS CLI peut être plus rapide et plus pratique que celle de la console. Les outils de ligne de commande sont également utiles si vous souhaitez créer des scripts exécutant les tâches AWS.

AWS fournit deux jeux d'outils de ligne de commande : l'AWS Command Line Interface (AWS CLI) et les Outils AWS pour Windows PowerShell. Pour plus d'informations sur l'installation et l'utilisation des AWS CLI, consultez le AWS Command Line Interface Guide de l'utilisateur. Pour plus d'informations sur l'installation et l'utilisation des Outils pour Windows PowerShell, consultez le Outils AWS pour Windows PowerShell Guide de l'utilisateur.

> **Note**
>
> Sur vos Windows Server instances, Windows PowerShell 3.0 ou version ultérieure est nécessaire pour exécuter certains documents SSM (par exemple, le `AWS-ApplyPatchBaseline` document hérité). Vérifiez que vos Windows Server instances sont en cours d'exécution Windows Management Framework 3.0 ou version ultérieure. inclut Windows PowerShell.

AWS SDKs

AWSfournit des kits SDK composés de bibliothèques et d'exemples de code pour différents langages et plateformes de programmation (par exemple, Java, Python, Ruby, .NET, iOS et Android,

etc.). permet de créer facilement un accès par programme à SDKs .Systems Manager Pour plus d'informations sur l' AWS SDKs , y compris la façon de les télécharger et de les installer, consultez Outils pour Amazon Web Services.

Prérequis d'Systems Manager

Les conditions préalables à l'utilisation d'AWS Systems Manager pour gérer vos instances EC2, les serveurs sur site et les machines virtuelles (VM), sont présentées étape par étape dans les chapitres Configuration de ce guide de l'utilisateur.

- Configuration d'AWS Systems Manager (p. 28)
- Configuration d'AWS Systems Manager pour les environnements hybrides (p. 49)

Cette rubrique fournit une présentation de ces conditions préalables.

Pour remplir les prérequis pour l'utilisation d'Systems Manager

1. Créez un AWS compte et configurez les rôles AWS Identity and Access Management (IAM) requis.
2. Vérifiez quSystems Manager' est pris en charge dans les AWS régions où vous souhaitez utiliser le service.
3. Vérifiez que vos instances exécutent un système d'exploitation pris en charge.
4. Pour les instances EC2, créez un profil dIAM'instance et attachez-le à vos machines.
5. Pour les serveurs sur site et VMs , créez un rôle de IAM service pour un environnement hybride.
6. (Recommandé) Créer un point de terminaison d'un VPC dans Amazon Virtual Private Cloud à utiliser avec Systems Manager.

   Si vous n'utilisez pas de VPC, vous devez configurer vos instances gérées pour autoriser `HTTPS` (port 443) le trafic sortant vers les Systems Manager points de terminaison. Pour plus d'informations, consultez (Facultatif) Créer un point de terminaison de VPC. (p. 42)
7. Sur les serveurs sur site, les et les instances EC2 créés à partir dVMs' qui ne sont pas fournis par AMIs , assurez-vous queAWS Transport Layer Security (TLS) est installé.
8. Pour les serveurs sur site et VMs , enregistrez les machines avec Systems Manager via le processus d'activation d'instance gérée.
9. Installez ou vérifier l'installation de l'Agent SSM sur chacune de vos instances gérées.

Intégration à IAM et Amazon EC2

L'accès utilisateur à Systems Manager, ses fonctionnalités et ses ressources sont contrôlées via des stratégies que vous utilisez ou créez dans AWS Identity and Access Management (IAM). Si vous prévoyez d'utiliser des ressources de calcul fournies par AWS, et pas uniquement des machines virtuelles et des serveurs sur site, vous devez également comprendre Amazon Elastic Compute Cloud (Amazon EC2) avant de configurer Systems Manager pour votre organisation. Il est essentiel de comprendre la façon dont ces services fonctionnent pour configurer Systems Manager avec succès.

Pour plus d'informations sur Amazon EC2, consultez les ressources suivantes :

- Amazon Elastic Compute Cloud (Amazon EC2)
- Mise en route avec les instances Amazon EC2 Linux
- Mise en route avec les instances Amazon EC2 Windows
- Présentation du Amazon EC2 (Linux)
- Présentation du Amazon EC2 Windows

Pour plus d'informations sur IAM, consultez les ressources suivantes :

- AWS Identity and Access Management (IAM)
- Démarrer avec IAM
- En quoi consiste IAM ?

AWS Systems Manager Configuration rapide

Utilisez AWS Systems Manager Configuration rapide pour configurer rapidement les rôles de sécurité requis et les Systems Manager fonctionnalités couramment utilisées sur vos Amazon EC2 instances. Vous pouvez utiliser Configuration rapide dans un compte individuel ou dans plusieurs comptes et régions en l'intégrant à AWS Organizations . Ces fonctionnalités vous aident à gérer et à surveiller l'état de vos instances tout en fournissant les autorisations minimales requises pour démarrer. Plus précisément, vous Configuration rapide aide à configurer les composants suivants sur les instances que vous choisissez ou ciblez à l'aide de balises :

- Rôles de profil d'instanceAWS Identity and Access Management (IAM) pour Systems Manager.
- Une mise à jour planifiée bihebdomadaire de Agent SSM.
- Une collecte planifiée de métadonnées d'inventaire toutes les 30 minutes.
- Une analyse quotidienne de vos instances pour identifier les correctifs manquants.
- Une installation et une configuration uniques de l'agent Amazon CloudWatch.
- Une mise à jour mensuelle planifiée de l'agent CloudWatch.

> **Note**
>
> Si votre organisation Configuration rapide cible un compte qui a précédemment exécuté une locale Configuration rapide , les configurations existantes ne sont pas modifiées. Un nouvel ensemble d'State Managerassociations est créé lorsque vous exécutez une organisationConfiguration rapide. Cela signifie que vous pouvez avoir des associations dont les options de configuration et les planifications se chevauchent.

L'organisation Configuration rapide est disponible dans les AWS régions suivantes :

- USA Est (Virginie du Nord)
- USA Est (Ohio)
- USA Ouest (Californie du Nord)
- USA Ouest (Oregon)
- Asie-Pacifique (Mumbai)
- Asie-Pacifique (Séoul)
- Asie-Pacifique (Singapour)
- Asie-Pacifique (Sydney)
- Asie-Pacifique (Tokyo)
- Europe (Francfort)
- Europe (Irlande)
- Europe (Londres)
- Europe (Paris)
- Canada (Centre)

- Amérique du Sud (São Paulo)

Local Configuration rapide est disponible dans toutes les régions où Systems Manager est pris en charge. Pour obtenir la liste des régions prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services .

Pour accéder à Configuration rapide , choisissez Configuration rapide dans le volet de navigation de la Systems Manager console . Vous pouvez également accéder Configuration rapide à en choisissant AWS Systems Manager en haut du volet de navigation, puis en choisissant Mise en route avec Systems Manager , comme illustré dans l'image suivante. Pour accéder au type Configuration rapide d'organisation, qui vous permet de cibler plusieurs comptes et régions, vous devez être connecté à l' compte principal de votre organisation. Pour plus d'informations sur la façon de démarrer avec AWS Organizations , consultez Mise en route avec AWS Organizations dans le Manuel de l'utilisateur AWS Organizations .

Note

Vous pouvez modifier Configuration rapide les configurations de votre compte à tout moment. Pour modifier une configuration d'Configuration rapideorganisation, l'état de la configuration doit indiquer Success (Réussite) ou Failed (Échec). Avant de modifier une Configuration rapide configuration, nous vous recommandons d'apprendre à modifier les configurations à l'aide de la page Configuration rapide Résultats. Pour plus d'informations, consultez la section Utilisation des Configuration rapide résultats (p. 24).

Rôles d'autorisations

Par défaut, Systems Manager n'a pas l'autorisation de communiquer avec vos instances ou d'effectuer des actions sur celles-ci. Vous devez lui accorder l'accès à l'aide d'un profil d'instance AWS Identity and Access Management (IAM) et d'un rôle de service IAM (ou d'un rôle de responsable ou assume role). Un profil d'instance est un conteneur qui transmet des informations sur le rôle IAM à une instance EC2 lors du lancement. Un rôle de service permet à Systems Manager d'exécuter des commandes sur vos instances. Pour de plus amples informations sur les profils d'instance, veuillez consulter Utilisation des profils d'instance dans le IAM Guide de l'utilisateur. Pour de plus amples informations sur les rôles de service, veuillez consulter Création d'un rôle pour déléguer des autorisations à un service AWS.

Vous pouvez décider d'utiliser ces rôles Configuration rapide pour qu'ils soient créés et configurés pour vous en choisissant Use the default role (Utiliser le rôle par défaut). Si vous sélectionnez un rôle existant, ce rôle doit inclure des stratégies IAM avec, au minimum, les autorisations décrites dans cette rubrique. Si vous sélectionnez des rôles existants et qu'ils ne disposent pas de ces autorisations, Configuration

rapide la configuration d'un ou de plusieurs composants sélectionnés risque d'échouer ou ces composants risquent de ne pas fonctionner correctement.

> **Note**
>
> Configuration rapidene remplace pas les profils d'instance qui existent déjà sur vos instances.

Détails sur le profil d'instance par défaut

Dans la section Rôle de profil d'instance, si vous choisissez Utiliser le rôle par défaut, Configuration rapide crée un profil dIAM'instance qui utilise la AmazonSSMManagedInstanceCore stratégie et une stratégie supplémentaire. La AmazonSSMManagedInstanceCore stratégie permet Systems Manager à d'effectuer les actions suivantes sur vos instances .

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeAssociation",
                "ssm:GetDeployablePatchSnapshotForInstance",
                "ssm:GetDocument",
                "ssm:DescribeDocument",
                "ssm:GetManifest",
                "ssm:GetParameter",
                "ssm:GetParameters",
                "ssm:ListAssociations",
                "ssm:ListInstanceAssociations",
                "ssm:PutInventory",
                "ssm:PutComplianceItems",
                "ssm:PutConfigurePackageResult",
                "ssm:UpdateAssociationStatus",
                "ssm:UpdateInstanceAssociationStatus",
                "ssm:UpdateInstanceInformation"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssmmessages:CreateControlChannel",
                "ssmmessages:CreateDataChannel",
                "ssmmessages:OpenControlChannel",
                "ssmmessages:OpenDataChannel"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2messages:AcknowledgeMessage",
                "ec2messages:DeleteMessage",
                "ec2messages:FailMessage",
                "ec2messages:GetEndpoint",
                "ec2messages:GetMessages",
                "ec2messages:SendReply"
            ],
            "Resource": "*"
        }
    ]
}
```

> **Note**
>
> Pour de plus amples informations sur les actions `ec2messages*` et `ssmmessages*`, veuillez consulter Référence : ec2messages, ssmmessages et autres appels d'API (p. 1399).

ajoute Configuration rapide également la stratégie suivante au profil d'instance. Cette stratégie autorise les communications approuvées entre le service Systems Manager dans le cloud et vos instances Amazon EC2.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Principal":{
            "Service":"ec2.amazonaws.com"
         },
         "Action":"sts:AssumeRole"
      }
   ]
}
```

Détails sur le rôle de service

Dans la section service role (Rôle de Systems Manager service), si vous choisissez Use the default role (Utiliser le rôle par défaut), crée un nouveau rôle Configuration rapide de service qui inclut les stratégies suivantes.IAM La première stratégie permet à Systems Manager d'effectuer les actions suivantes sur vos instances.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iam:CreateInstanceProfile",
                "iam:ListInstanceProfilesForRole",
                "iam:PassRole",
                "ec2:DescribeIamInstanceProfileAssociations",
                "iam:GetInstanceProfile",
                "ec2:DisassociateIamInstanceProfile",
                "ec2:AssociateIamInstanceProfile",
                "iam:AddRoleToInstanceProfile"
            ],
            "Resource": "*"
        }
    ]
}
```

La stratégie suivante permet à Systems Manager d'effectuer les actions de la stratégie précédente en votre nom. Systems Manager endosse votre rôle pour effectuer les actions.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Principal":{
            "Service":"ssm.amazonaws.com"
         },
```

```
            "Action":"sts:AssumeRole"
        }
    ]
}
```

> **Note**
>
> La configuration d'une instance avec un profil d'instance pour Systems Manager n'autorise pas un utilisateur à exécuter des commandes ou à utiliser des fonctionnalités Systems Manager sur cette instance. Votre utilisateur, groupe ou rôle IAM doit être configuré avec une stratégie d'autorisations distincte qui vous permet d'exécuter des actions sur vos instances à l'aide de Systems Manager. Pour plus d'informations, consultez la section Configuration d'AWS Systems Manager (p. 28).

Mettre à jour l'agent Systems Manager (SSM)

Agent SSM est un logiciel Amazon qui traite les demandes du service Systems Manager dans le cloud AWS, puis les exécute sur votre instance comme indiqué dans la demande. Agent SSM est préinstallé, par défaut, sur les Amazon Machine Images (AMI) suivantes :

- Amazon Linux
- Amazon Linux 2
- AMIs optimisées pour ECS Amazon Linux 2
- Ubuntu Server 16.04, 18.04 et 20.04
- AMIs Windows Server 2008-2012 R2 publiées à partir de novembre 2016
- Windows Server 2016 et 2019

> **Note**
>
> L'Agent SSM n'est pas installé sur toutes les AMIs basées sur Amazon Linux ou Amazon Linux 2. Par exemple, l'Agent SSM n'est pas préinstallé sur les AMIs optimisées pour EKS basées sur Amazon Linux 2.

Si vous activez cette option, Systems Manager recherche automatiquement toutes les deux semaines une nouvelle version de l'agent. S'il existe une nouvelle version, Systems Manager met automatiquement à jour l'agent sur votre instance vers la dernière version publiée. Nous vous encourageons à choisir cette option pour être sûr que vos instances exécutent toujours la version la plus récente de l'Agent SSM. Pour de plus amples informations sur Agent SSM, y compris sur la façon d'installer manuellement l'agent, veuillez consulter Utilisation des Agent SSM (p. 72).

Collecter l'inventaire à partir de vos instances

AWS Systems Manager Inventory vous permet d'avoir une certaine visibilité sur votre environnement informatique. Vous pouvez utiliser l'inventaire pour collecter les métadonnées à partir de vos instances gérées. Vous pouvez stocker ces métadonnées dans un compartiment Amazon Simple Storage Service (Amazon S3) central. Ensuite, utilisez les outils intégrés pour interroger les données et déterminer rapidement quelles instances exécutent les logiciels et les configurations requis par votre stratégie logicielle, et quelles instances doivent être mises à jour. Configuration rapide configure la collecte pour les types de métadonnées suivants :

- Composants AWS : Pilote EC2, agents, versions, etc.
- Applications : Noms des applications, éditeurs, versions, etc.

- Détails de l'instance : Nom du système, nom du système d'exploitation, version du système d'exploitation, dernier démarrage, DNS, domaine, groupe de travail, architecture du système d'exploitation, etc.
- Configuration réseau : Adresse IP, adresse MAC, DNS, passerelle, masque de sous-réseau, etc.
- Services : Nom, nom d'affichage, statut, services dépendants, type de service, type de démarrage, etc. (Windows Serverinstances uniquement).
- Rôles Windows : Nom, nom d'affichage, chemin, type de fonction, état installé, etc. (Windows Serverinstances uniquement).
- Mises à jour Windows : ID du correctif, installé par, date d'installation, etc. (Windows Serverinstances uniquement).

Vous pouvez configurer Systems Manager Inventory pour collecter les types de métadonnées supplémentaires suivants à partir de vos instances. Pour plus d'informations, consultez la section Inventaire AWS Systems Manager (p. 791).

- Inventaire personnalisé : Métadonnées affectées à une instance gérée comme décrit dans Utilisation de l'inventaire personnalisé (p. 826) .
- Fichiers : Nom, taille, version, date d'installation, heure de modification, heure du dernier accès, etc.
- Balises : Balises attribuées à vos instances.
- Registre Windows : Chemin de la clé de registre, nom de valeur, type de valeur et valeur.

> **Important**
>
> Une configuration d'inventaire (Inventory) globale est créée à partir de l'option de configuration en un clic sur la page Inventory (Inventaire) et cible toutes les instances de votre compte AWS. Si une configuration d'inventaire globale est déjà activée sur vos instances, l'activation de la collecte d'inventaire dans la configuration rapide crée une autre planification de collecte des données d'inventaire pour les instances spécifiées ou ciblées. Il n'y a pas de conflit. Toutefois, si une configuration d'inventaire non globale est déjà activée sur certaines de vos instances, la configuration que vous avez créée précédemment échouera. La collecte d'inventaire ne prend pas en charge plus d'une configuration spécifique par instance. Pour résoudre ce problème, déployez une configuration d'inventaire globale ou supprimez la configuration à l'origine du conflit.

Analyser quotidiennement les instances pour rechercher les correctifs manquants

Si vous activez cette option dans Configuration rapide , Systems Manager utilise Gestionnaire de correctifs pour analyser vos instances chaque jour et générer un rapport simple dans la page Conformité. Le rapport indique combien d'instances sont conformes aux correctifs en fonction de la référence de correctifs par défaut. Le rapport inclut une liste de chaque instance et de son statut de conformité. Vous pouvez parcourir cette liste pour afficher des détails sur les instances non conformes. Pour de plus amples informations sur les opérations d'application de correctifs et les références de correctifs, veuillez consulter AWS Systems Manager Gestionnaire de correctifs (p. 1047). Pour afficher les informations de conformité, veuillez consulter la page Systems Manager Conformité .

Installer et configurer l'agent CloudWatch

Amazon CloudWatch met à votre disposition des données et des informations exploitables pour surveiller vos applications, comprendre les variations de performances systémiques et y répondre, optimiser l'utilisation des ressources et obtenir une vue unifiée de la santé opérationnelle. L'agent

CloudWatch collecte des métriques et des fichiers journaux à partir de vos instances et consolide ces informations afin que vous puissiez rapidement déterminer l'état de vos instances. Pour plus d'informations, consultez Collecte de métriques et de journaux à partir d'instances EC2 et de serveurs sur site avec lCloudWatch'agent . Des frais supplémentaires peuvent vous être facturés. Pour de plus amples informations, veuillez consulter Tarification de Amazon CloudWatch.

Mettre à jour l'agent CloudWatch toutes les quatre semaines

Si vous activez cette option, Systems Manager recherche automatiquement une nouvelle version de l'agent CloudWatch toutes les quatre semaines. S'il existe une nouvelle version, Systems Manager met automatiquement à jour l'agent sur votre instance vers la dernière version publiée. Nous vous encourageons à choisir cette option pour être sûr que vos instances exécutent toujours la version la plus récente de l'agent CloudWatch.

Choix de cibles pourConfiguration rapide

Une fois que vous avez choisi Configuration rapide les options, choisissez les instances que vous souhaitez configurer avec ces options dans la section Targets (Cibles). Configuration rapide inclut les options suivantes pour cibler les instances.

Local

- Choisissez toutes les instances du compte et de la région AWS actuels : Configuration rapide localise et applique les options de configuration à toutes les instances du compte et de la région AWS actuels.
- Specify instance tags: Configuration rapide utilise la clé de balise et la valeur de balise (facultative) que vous spécifiez pour localiser les instances.
- Choose instances manually (Choisir des instances manuellement) : vous Configuration rapide permet de choisir une ou plusieurs instances dans une liste.

Organization

- Les unités d'organisation cibles : vous Configuration rapide permet de cibler plusieurs unités dAWS Organizations'organisation. Configuration rapide applique votre configuration à tous les comptes au sein d'une unité d'organisation.
- Région cible : vous Configuration rapide permet de choisir les régions à cibler au sein de vos unités organisationnelles cibles.

> **Note**
>
> Vous ne pouvez pas cibler des instances individuelles ou des balises lors de l'utilisation de l'organisationConfiguration rapide. Toutes les instances dans les comptes au sein des unités d'organisation cibles et de la région cible sont configurées par Configuration rapide . applique Configuration rapide également votre configuration à toutes les nouvelles instances que vous lancez dans les comptes au sein de vos unités d'organisation cibles et de votre région cible. Vous pouvez spécifier jusqu'à 2 000 cibles avec OrganizationConfiguration rapide. Les cibles d'une organisation correspondent aux instances de pile.Configuration rapideAWS CloudFormation Cela signifie qu'une cible est une référence à une pile dans un compte au sein d'une région. Vous pouvez obtenir le nombre total de cibles en multipliant le nombre de comptes par le nombre de régions spécifié. Pour plus d'informations sur les instances AWS CloudFormation de piles, consultez Instances de piles dans le AWS CloudFormation Guide de l'utilisateur .

Utilisation des Configuration rapide résultats

Systems Manageraffiche les résultats de Configuration rapide dans une carte distincte pour chaque option que vous avez sélectionnée.

Pour chaque option sélectionnée, Systems Manager crée et exécute immédiatement une association State Manager. Le champ Configuration status (État de la configuration) affiche Success (Succès) lorsque l'association s'exécute correctement sur toutes les instances sélectionnées ou ciblées. Si l'exécution d'une association échoue, le champ Configuration status (État de la configuration) affiche Failed (Échec). Si une association est en cours de traitement, le champ Configuration status (État de la configuration) affiche Pending (En attente). Pour mettre à jour l'état de configuration des éléments en attente, actualisez votre navigateur.

> **Note**
>
> L'exécution de l'option Inventory collection (Collecte d'inventaire) peut prendre jusqu'à 10 minutes pour s'exécuter, même si vous n'avez sélectionné que quelques instances.

Un état Configuration Non configuré signifie que vous n'avez pas choisi l'option dans .Configuration rapide Si vous voyez cet état affiché pour Managed instances (Instances gérées), cela signifie que vous n'avez pas choisi de rôle dans la liste Role selection (Sélection de rôle). Vous pouvez exécuter Configuration rapide à nouveau, comme décrit dans cette rubrique, et choisir un rôle.

Pour modifier l'association d'une Configuration rapide option, choisissez le bouton Modifier dans la carte d'option. Si vous modifiez l'association, ne choisissez pas un autre document SSM dans la page Modifier l'association. Si vous choisissez un autre document SSM, l'option devient indisponible dans Configuration rapide . Modifiez uniquement les paramètres et les cibles de l'association. Lorsque vous enregistrez vos modifications dans l'association, State Manager exécute automatiquement l'association.

Pour afficher les données de conformité pour votre Configuration rapide , choisissez le bouton View compliance summary in Explorer (Afficher le récapitulatif de conformité dans l'Explorateur). Cette opération permet d'ouvrir le AWS Systems Manager Explorer tableau de bord. Pour plus d'informations sur Explorer, consultez AWS Systems Manager Explorer (p. 153).

Lorsque vous consultez Configuration rapide les résultats lorsque vous êtes connecté à compte principal pour votre AWS Organizations organisation, vous voyez un résumé des résultats de votre organisation Configuration rapide dans l'onglet Organisation. Cette vue affiche la configuration appliquée pour votre Configuration rapide , vous permet de filtrer les résultats, affiche les statuts de déploiement et d'association pour vos cibles et vous permet d'afficher plus de détails sur les cibles individuelles dans la carte de Configuration rapide déploiements.

Le statut de déploiement fait référence à l'état d'une cible ou d'une instance de pile. En d'autres termes, cet état indique si AWS CloudFormation a déployé avec succès vos options de configuration. Le statut du déploiement n'évalue pas le statut des associations créées dans votre Configuration rapide .

L'état d'association fait référence à l'état de toutes les associations créées par votre Configuration rapide au sein d'une cible ou d'une instance de pile. Toutes les associations doivent être exécutées avec succès, sinon le statut de la cible est Failed.

Configuration rapideDépannage des résultats

Si une Configuration rapide carte indique Non configuré, vous avez peut-être oublié une option sur la Configuration rapide page. Pour résoudre ce problème, choisissez le bouton Modifier tout en haut de la page des Configuration rapide résultats et passez en revue vos sélections. Si vous avez oublié une ou plusieurs options, vous pouvez les sélectionner, puis choisir Réinitialiser pour les configurer.

Si vous voyez toujours un problème avec une ou plusieurs cartes de Configuration rapide résultats, utilisez la procédure suivante pour résoudre le problème.

> **Note**
>
> Pour examiner les associations ayant échoué pour une organisationConfiguration rapide, vous devez vous connecter au compte avec l'association ayant échoué et utiliser la procédure suivante. L'ID d'association n'est pas un lien hypertexte vers le compte de destination lors de l'affichage des résultats à partir de l' compte principal .

Pour dépanner une Configuration rapide configuration ayant échoué

1. Dans la page Configuration rapide des résultats, choisissez View Details in the card with a Configuration status of Failed.

2. Sur la page Association ID (ID d'association), choisissez l'onglet Execution history (Historique d'exécution).
3. Sous Execution ID (ID d'exécution), choisissez l'exécution d'association qui a échoué.

4. La page Association execution targets (Cibles d'exécution d'association) répertorie toutes les instances sur lesquelles l'association s'est exécutée. Choisissez le bouton Output (Sortie) pour une exécution dont l'exécution a échoué.

5. Sur la page Output (Sortie), choisissez Step - Output (Étape - Sortie) pour afficher le message d'erreur pour cette étape dans l'exécution de la commande. Chaque étape peut afficher un message d'erreur différent. Passez en revue les messages d'erreur de toutes les étapes afin de faciliter la résolution du problème.

Si l'affichage de la sortie de l'étape ne résout pas le problème, vous pouvez recréer l'association pour voir si le problème persiste. Pour recréer l'association, vous devez d'abord supprimer toutes les associations qui existent déjà pour l'Configuration rapideoption. Vous pouvez supprimer une association à l'aide du bouton Supprimer de la carte Configuration rapide des résultats. Vous pouvez également supprimer l'association en choisissant State Manager dans le panneau de navigation. Une fois que vous avez supprimé l'association, exécutez à Configuration rapide nouveau pour voir si le problème persiste.

En cours d'exécution Configuration rapide à nouveau

Vous pouvez exécuter Configuration rapide à nouveau en choisissant le bouton Edit all (Modifier tout) sur la page Configuration rapide des résultats. Vous pouvez activer ou désactiver les options. Si vous désactivez une option, Systems Manager ne supprime pas l'association. Par exemple, si vous avez déjà sélectionné l'option Collect inventory from your instances every 30 minutes (Collecter l'inventaire à partir de vos instances toutes les 30 minutes) et que vous désactivez l'option lorsque vous Configuration rapide exécutez à nouveau, l'association d'origine existe toujours. Vous devez supprimer l'association à partir de la page Configuration rapide des résultats ou de la page State Manager.

> **Important**
>
> Si vous souhaitez exécuter Configuration rapide sur de nouvelles instances, nous vous recommandons de choisir les nouvelles instances, puis de choisir toutes les instances sur lesquelles vous avez précédemment exécuté la configuration rapide. En choisissant toutes les instances sur lesquelles vous avez précédemment exécuté Configuration rapide , vous synchronisez les exécutions d'association pour toutes les instances. Cela permet de synchroniser les rapports d'état et de conformité. Nous vous recommandons également de cibler les instances à l'aide de balises. Les nouvelles instances avec les balises spécifiées sont automatiquement ajoutées aux Configuration rapide associations. Cela signifie qu'ils affichent automatiquement le statut dans les Configuration rapide résultats et sur la page Conformité.

Configuration d'AWS Systems Manager

Cette section décrit les tâches effectuées par les administrateurs système et de compte pour configurer AWS Systems Manager pour leurs organisations. Une fois ces étapes terminées, les utilisateurs de l'organisation peuvent utiliser Systems Manager pour configurer et gérer les instances Amazon Elastic Compute Cloud (EC2) dans leur compte.

Si vous prévoyez d'utiliser Systems Manager pour gérer et configurer vos propres serveurs et machines virtuelles sur site dans un environnement hybride, suivez les étapes décrites dans Configuration d'AWS Systems Manager pour les environnements hybrides (p. 49). Si vous prévoyez d'utiliser des instances EC2 et vos propres ressources informatiques dans un environnement hybride, commencez par suivre les étapes décrites ici. Cette section présente les étapes dans l'ordre le plus adapté pour configurer les rôles, les utilisateurs, les autorisations et les ressources initiales à utiliser dans vos opérations Systems Manager.

Si vous utilisez déjà d'autres services AWS, vous avez effectué certaines de ces étapes. Cependant, d'autres étapes sont spécifiques à Systems Manager. Par conséquent, nous vous recommandons de passer en revue l'intégralité de cette section afin de vérifier que vous êtes prêt à utiliser toutes les fonctionnalités Systems Manager.

> Note
>
> Vous pouvez utiliser la configuration rapide Systems Manager pour configurer rapidement un profil d'instance AWS Identity and Access Management (IAM) sur toutes les instances de votre compte AWS. La configuration rapide peut également créer un rôle de type assume role, ce qui permet à Systems Manager d'exécuter en toute sécurité des commandes sur vos instances en votre nom. La configuration rapide vous permet d'ignorer les étapes 4 et 5 de cette section. Pour plus d'informations, consultez la section AWS Systems Manager Configuration rapide (p. 17).

Table des matières

- Étape 1 : Inscrivez-vous à AWS (p. 28)
- Étape 2 : Créer un utilisateur IAM administrateur pour AWS (p. 29)
- Étape 3 : Créer des utilisateurs et des groupes IAM non administrateurs pour Systems Manager (p. 30)
- Étape 4 : Créer un profil dIAM'instance pourSystems Manager (p. 34)
- Étape 5 : Attachement d'un profil d'instance IAM à une instance EC2 (p. 40)
- Étape 6 : (Facultatif) Créer un point de terminaison de VPC (p. 42)
- Étape 7 : (Facultatif) Créer des rôles de service Systems Manager (p. 45)
- Étape 8 : (Facultatif) Configuration des intégrations avec d'autres AWS services (p. 47)

Étape 1 : Inscrivez-vous à AWS

Si vous n'avez pas de compte AWS, suivez la procédure suivante pour en créer un.

Pour s'inscrire sur AWS

1. Ouvrez https://aws.amazon.com/, puis choisissez Create an AWS Account.

2. Suivez les instructions en ligne.

Passez au Étape 2 : Créer un utilisateur IAM administrateur pour AWS (p. 29).

Étape 2 : Créer un utilisateur IAM administrateur pour AWS

Lorsque vous créez un compte AWS, vous commencez avec une seule identité de connexion disposant d'un accès complet à tous les services et ressources AWS du compte. Cette identité est appelée la utilisateur racinedu compte AWS et elle est accessible après connexion à l'aide de l'adresse e-mail et du mot de passe utilisés pour la création du compte. Il est vivement recommandé de ne pas utiliser l'utilisateur racine pour vos tâches quotidiennes, y compris pour les tâches administratives. Au lieu de cela, respectez la bonne pratique qui consiste à avoir recours à l'utilisateur racine uniquement pour créer le premier utilisateurIAM. Ensuite, mettez en sécurité les informations d'identification de l'utilisateur racine et utilisez-les pour effectuer uniquement certaines tâches de gestion des comptes et des services.

Dans cette procédure, vous utilisez Utilisateur racine d'un compte AWS pour créer votre premier utilisateur dans AWS Identity and Access Management (IAM). Vous ajoutez cet utilisateur IAM à un groupe d'administrateurs, afin d'être sûr d'avoir accès à l'ensemble des services et de leurs ressources dans votre compte. La prochaine fois que vous accèderez à votre compte AWS, connectez-vous avec les informations d'identification de cet utilisateur IAM.

Pour créer un utilisateur IAM avec des autorisations restreintes, veuillez consulter Étape 3 : Créer des utilisateurs et des groupes IAM non administrateurs pour Systems Manager (p. 30).

Pour créer un administrateur pour vous-même et ajouter l'utilisateur à un groupe d'administrateurs (console)

1. Connectez-vous à la console IAM en tant que propriétaire du compte en choisissant utilisateur racine et en saisissant l'adresse e-mail de votre compte AWS. Sur la page suivante, saisissez votre mot de passe.

   Note

   Nous vous recommandons vivement de respecter la bonne pratique qui consiste à avoir recours à l'utilisateur **`Administrator`** IAM ci-dessous et à mettre en sécurité les informations d'identification de l'utilisateur racine. Connectez-vous en tant qu'utilisateur racine pour effectuer certaines tâches de gestion des comptes et des services.
2. Dans le panneau de navigation, choisissez Utilisateurs, puis Add user (Ajouter un utilisateur).
3. Dans User name (Nom d'utilisateur), entrez **`Administrator`**.
4. Cochez la case en regard de AWS Management Console access. Puis, sélectionnez Custom password (Mot de passe personnalisé, et entrez votre nouveau mot de passe dans la zone de texte.
5. Par défaut, AWS oblige le nouvel utilisateur à créer un nouveau mot de passe lors de sa première connexion. Décochez la case en regard de User must create a new password at next sign-in (L'utilisateur doit créer un nouveau mot de passe à sa prochaine connexion) pour autoriser le nouvel utilisateur à réinitialiser son mot de passe une fois qu'il s'est connecté.
6. Choisissez Next: Permissions (Suivant : Autorisations).
7. Sous Set permissions (Accorder des autorisations), choisissez Add user to group (Ajouter un utilisateur au groupe).
8. Choisissez Create group.
9. Dans la boîte de dialogue Create group (Créer un groupe), pour Group name (Nom du groupe), tapez **`Administrators`**.

10. Choisissez Filter policies (Filtrer les stratégies), puis sélectionnez AWS managed -job function (Fonction -job gérée par) pour filtrer le contenu de la table.
11. Dans la liste des stratégies, cochez la case AdministratorAccess. Choisissez ensuite Create group.

    Note

    Vous devez activer l'accès des rôles et utilisateurs IAM à la facturation avant de pouvoir utiliser les autorisations `AdministratorAccess` pour accéder à la console AWS Billing and Cost Management. Pour ce faire, suivez les instructions de l'étape 1 du didacticiel portant sur comment déléguer l'accès à la console de facturation.
12. De retour dans la liste des groupes, activez la case à cocher du nouveau groupe. Choisissez Refresh si nécessaire pour afficher le groupe dans la liste.
13. Choisissez Next: Tags (Suivant : Balises).
14. (Facultatif) Ajoutez des métadonnées à l'utilisateur en associant les balises sous forme de paires clé-valeur. Pour plus d'informations sur l'utilisation des balises dans IAM, consultez Balisage des entités IAM dans le IAM Guide de l'utilisateur.
15. Choisissez Next: Review pour afficher la liste des membres du groupe à ajouter au nouvel utilisateur. Une fois que vous êtes prêt à continuer, choisissez Create user.

Vous pouvez utiliser ce même processus pour créer d'autres groupes et utilisateurs et pour accorder à vos utilisateurs l'accès aux ressources de votre compte AWS. Pour en savoir plus sur l'utilisation des stratégies afin de limiter les autorisations d'accès des utilisateurs à certaines ressources AWS, consultez Gestion des accès et Exemples de stratégies.

Passez au Étape 3 : Créer des utilisateurs et des groupes IAM non administrateurs pour Systems Manager (p. 30).

Étape 3 : Créer des utilisateurs et des groupes IAM non administrateurs pour Systems Manager

Les utilisateurs du groupe d'administrateurs d'un compte ont accès à l'ensemble des services et des ressources AWS de ce compte. Cette section décrit comment créer des utilisateurs avec des autorisations limitées à AWS Systems Manager.

Les fonctions Systems Manager suivantes peuvent comporter des procédures complémentaires ou différentes pour accorder un accès utilisateur :

- Session Manager - Consultez Contrôler les accès de session utilisateur aux instances (p. 882).
- Distributor - Consultez Contrôle de l'accès des utilisateurs aux packages (p. 1179).
- Fenêtres de maintenance- Consultez Configuration d'Fenêtres de maintenance (p. 690) (consultez les instructions relatives à l'attribution de la IAM PassRole stratégie à un IAM utilisateur ou un groupe).

Pour de plus amples informations sur l'utilisation des stratégies IAM pour contrôler l'accès des utilisateurs aux fonctions et ressources Systems Manager, veuillez consulter Exemples de stratégies basées sur l'identité AWS Systems Manager (p. 1304).

Pour de plus amples informations sur la modification des autorisations pour un compte utilisateur, un groupe ou un rôle IAM, veuillez consulter Modification des autorisations pour un utilisateur IAM dans le IAM Guide de l'utilisateur.

Rubriques

- Tâche 1 : Créer des groupes d'utilisateurs (p. 31)

- Tâche 2 : Créer des utilisateurs et attribuer des autorisations (p. 32)

Tâche 1 : Créer des groupes d'utilisateurs

Vous pouvez créer un groupe d'utilisateurs pour chaque stratégie et attribuer des utilisateurs à un groupe, plutôt que d'attacher des stratégies individuelles à chaque utilisateur.

Vous pouvez créer plusieurs groupes d'utilisateurs avec différents jeux d'autorisations en omettant les stratégies recommandées ou facultatives. Vous pouvez également créer des stratégies IAM personnalisées pour accorder n'importe quelle combinaison d'autorisations à un utilisateur. Par exemple, vous pouvez accorder à un groupe d'utilisateurs l'autorisation d'utiliser uniquement la fonction Session Manager dans Systems Manager, comme décrit dans Contrôler les accès de session utilisateur aux instances (p. 882).

Pour obtenir des exemples supplémentaires de stratégies IAM personnalisées pour Systems Manager, veuillez consulter Exemples de stratégies gérées par le client (p. 1307).

Pour obtenir des informations complètes sur l'utilisation de stratégies IAM pour l'accès à Systems Manager, veuillez consulter Identity and Access Management pour AWS Systems Manager (p. 1293) .

Pour créer un groupe d'utilisateurs

Utilisez la procédure suivante pour créer un groupe d'utilisateurs pour vos utilisateurs Systems Manager. Vous pouvez répéter cette procédure pour créer d'autres groupes d'utilisateurs avec différents jeux d'autorisations.

1. Dans le volet de navigation de la console IAM, choisissez Groupes, puis Créer un nouveau groupe.
2. Sur la page Définir un nom de groupe, entrez un nom pour le groupe, comme **`SSMUserGroup`**, ou tout autre nom de votre choix.
3. Choisissez Next Step.
4. Sur la page Attacher la stratégie, pour Filtre, entrez **`SSM`**.
5. Dans la liste des stratégies, procédez comme suit :

   - Pour fournir aux utilisateurs l'autorisation d'utiliser Resource Groups et Tag Editor, choisissez la ResourceGroupsandTagEditorFullAccess stratégie.

     Les groupes de ressources AWS peuvent être gérés dans le service Groupes de ressources AWS et dans la fonctionnalité Groupes de ressources dans Systems Manager. Il n'est pas obligatoire de fournir les utilisateurs et les groupes d'utilisateurs lors de l'accès de votre compte à ce service et à Tag Editor, mais nous vous recommandons de le faire pour rendre les opérations de gestion plus efficaces.

     Pour de plus amples informations, veuillez consulter Resource Groups dans AWS Systems Manager (p. 212).
   - Pour fournir aux utilisateurs de ce groupe un accès complet à la console Systems Manager, cochez la case en regard de `AmazonSSMFullAccess`.

     -ou-

     Si vous voulez que les utilisateurs de ce groupe puissent uniquement afficher des données Systems Manager sans pouvoir créer ou mettre à jour des ressources, cochez la case en regard de `AmazonSSMReadOnlyAccess`.
   - Pour fournir aux utilisateurs l'accès aux pages Built-In Insights (Analyses intégrées) et Dashboard by CloudWatch (Tableau de bord par) dans la console Systems Manager, cochez les cases en regard de ces stratégies gérées :
     - AWSHealthFullAccess

Cette stratégie accorde un accès complet aux notifications AWS Health APIs et , ainsi qu'à Personal Health Dashboard. Elle fournit également l'accès à des parties du tableau de bord des analyses intégrées dans la console Systems Manager.

- AWSConfigUserAccess

  Cette stratégie fournit un accès en lecture seule pour utiliser AWS Config, notamment pour faire des recherches par balise sur des ressources et lire toutes les balises. Elle fournit également l'accès à des parties du tableau de bord des analyses intégrées dans la console Systems Manager.

- CloudWatchReadOnlyAccess

  Cette stratégie fournit un accès en lecture seule à CloudWatch, qui est nécessaire pour afficher des informations sur le tableau de bord Dashboard by CloudWatch (Tableau de bord par) dans la console Systems Manager.

- Ajoutez les autres stratégies fournissant les autorisations que vous souhaitez accorder à ce groupe d'utilisateurs.

6. Choisissez Next Step.
7. Sur la page Vérifier, vérifiez que les bonnes stratégies sont ajoutées à ce groupe, puis choisissez Créer un groupe.

Passez au Tâche 2 : Créer des utilisateurs et attribuer des autorisations (p. 32).

Tâche 2 : Créer des utilisateurs et attribuer des autorisations

Créez des utilisateurs IAM pour les personnes ayant besoin d'accéder à AWS Systems Manager et ajoutez chaque utilisateur au groupe d'utilisateurs approprié afin qu'il dispose du bon niveau d'autorisation.

> **Note**
>
> Si votre organisation dispose déjà d'un système d'identité, vous souhaiterez peut-être créer une option d'authentification unique (single sign-on, ou SSO). L'authentification unique permet aux utilisateurs d'accéder à AWS Management Console pour votre compte même s'ils ne disposent pas d'une identité d'utilisateur IAM. Grâce à l'authentification unique, les utilisateurs n'ont plus non plus besoin de se connecter au site de votre organisation d'un côté et à AWS de l'autre. Pour de plus amples informations, veuillez consulter Activation de l'accès de broker d'identité personnalisé à la console AWS.

Selon que les comptes utilisateur ont déjà été créés ou non pour ce groupe, utilisez l'une des procédures suivantes :

Pour créer des utilisateurs et ajouter des autorisations

1. Dans le panneau de navigation de la console IAM, choisissez Utilisateurs, puis Ajouter un utilisateur.
2. Pour Nom d'utilisateur, entrez le nom que l'utilisateur indiquera pour se connecter à AWS Systems Manager.
3. Pour permettre à l'utilisateur d'accéder à l'API AWS, à l'AWS CLI, au kit SDK AWS, ainsi qu'aux autres outils de développement, cochez la case en regard de Programmatic access (Accès par programme).

   Cela crée une clé d'accès pour le nouvel utilisateur. Vous pouvez afficher ou télécharger les clés d'accès lorsque vous accédez à la page Final.
4. Pour permettre à l'utilisateur d'accéder à l'AWS Management Console, cochez la case en regard de Accès à AWS Management Console.

AWS Management Console fournit une interface web dans laquelle vous pouvez gérer vos ressources de calcul, de stockage et d'autres ressources du cloud. Au sein d'AWS Management Console, les services individuels ont leur propre console. Par exemple, vous pouvez gérer vos ressources de calcul à l'aide de la console Amazon EC2, et votre stockage à l'aide de la console Amazon S3.

Si vous choisissez Custom password (Mot de passe personnalisé), entrez un mot de passe initial pour l'utilisateur. Vous pouvez également activer la case à cocher Require password reset (Exiger une réinitialisation du mot de passe) pour obliger l'utilisateur à créer un nouveau mot de passe lors de sa prochaine connexion.

5. Choisissez Suivant : Autorisations.
6. Sur la page Set permissions for user (Définir les autorisations pour l'utilisateur), choisissez Add user to group (Ajouter un utilisateur au groupe).
7. Dans la liste des groupes, choisissez le groupe d'utilisateurs auquel ajouter l'utilisateur, puis cliquez sur Suivant : Balises.
8. (Facultatif) Ajoutez une ou plusieurs paires clé-valeur de balise pour organiser, suivre ou contrôler l'accès pour cet utilisateur, puis choisissez Suivant : Vérifiez pour voir la liste des appartenances à des groupes que le nouvel utilisateur rejoint.
9. Choisissez Create user (Créer un utilisateur).
10. Pour afficher les clés d'accès des utilisateurs (clé d'accès IDs et clés d'accès secrètes), choisissez Afficher en regard de chaque mot de passe et clé d'accès que vous souhaitez afficher. Pour enregistrer les clés d'accès, choisissez `Download .csv`, puis enregistrez le fichier dans un emplacement sûr.

    > **Important**
    >
    > C'est votre seule occasion de visualiser ou de télécharger les clés d'accès secrètes, et vous devez fournir ces informations à vos utilisateurs pour qu'ils puissent utiliser l'API AWS ou l'AWS CLI. Enregistrez les nouveaux ID de clé d'accès et clé d'accès secrète de l'utilisateur dans un endroit sûr et sécurisé. Vous ne pourrez plus accéder aux clés d'accès secrètes après cette étape.

11. Fournissez à chaque utilisateur ses informations d'identification. Sur la page finale, vous pouvez choisir Send email (Envoyer un e-mail) en regard de chaque utilisateur. Votre client de messagerie local s'ouvre avec un modèle que vous pouvez personnaliser et envoyer. Le modèle d'e-mail inclut les informations suivantes pour chaque utilisateur :

    - Nom utilisateur
    - URL de la page de connexion au compte. Utilisez l'exemple suivant, en remplaçant l'ID et l'alias de compte comme approprié :

    ```
    https://AWS-account-ID or alias.signin.aws.amazon.com/console
    ```

    Pour de plus amples informations, veuillez consulter Comment les utilisateurs IAM se connectent à AWS dans le IAM Guide de l'utilisateur.

    > **Important**
    >
    > Le mot de passe de l'utilisateur n'est pas inclus dans l'e-mail généré. Vous devez le fournir à l'utilisateur d'une manière conforme aux normes de sécurité de votre organisation.

Pour ajouter des autorisations pour un utilisateur existant

1. Dans le panneau de navigation de la console IAM, sélectionnez Utilisateurs.
2. Choisissez le nom de l'utilisateur à ajouter à un groupe, puis choisissez Ajouter autorisation.

3. Pour Ajouter un utilisateur aux groupes, cochez la case en regard du groupe auquel ajouter l'utilisateur, par exemple **`SSMUserGroup`**, ou du nom d'un autre groupe d'utilisateurs que vous avez créé.
4. Ajoutez les autres éventuelles stratégies d'autorisation disponibles à affecter à l'utilisateur.
5. Choisissez Suivant : Passez en revue pour voir la liste des appartenances à des groupes qui seront ajoutées au nouvel utilisateur.
6. Choisissez Add permissions (Ajouter des autorisations).

Passez au Étape 4 : Créer un profil dIAM'instance pourSystems Manager (p. 34).

Étape 4 : Créer un profil dIAM'instance pourSystems Manager

Par défaut, AWS Systems Manager n'a pas le droit d'effectuer des actions sur vos instances. Vous devez accorder l'accès en utilisant un profil d'instance AWS Identity and Access Management (IAM). Un profil d'instance est un conteneur qui transmet des informations sur le rôle IAM à une instance Amazon Elastic Compute Cloud (Amazon EC2) lors du lancement. Vous pouvez créer un profil d'instance pour Systems Manager en attachant une ou plusieurs stratégies IAM qui définissent les autorisations nécessaires pour un nouveau rôle ou un rôle que vous avez déjà créé.

> Note
>
> Vous pouvez utiliser la configuration rapide Systems Manager pour configurer rapidement un profil d'instance sur toutes les instances de votre compte AWS. La configuration rapide crée également un rôle de IAM service (ou rôle de responsable), qui permet Systems Manager à d'exécuter en toute sécurité des commandes sur vos instances en votre nom. La configuration rapide vous permet d'ignorer cette étape (Étape 4) et l'étape 5. Pour plus d'informations, consultez la section AWS Systems Manager Configuration rapide (p. 17).

Veuillez noter les détails suivants sur la création d'un profil d'instance IAM :

- Si vous configurez des serveurs ou des machines virtuelles dans un environnement hybride pour Systems Manager, vous n'avez pas besoin de leur créer de profil d'instance. Au lieu de cela, vous devez configurer vos serveurs et VMs pour utiliser un rôle IAM de service . Pour plus d'informations, consultez la section Créer un rôle de service IAM pour un environnement hybride (p. 50).
- Si vous modifiez le profil d'instance IAM, l'actualisation des informations d'identification de l'instance peut prendre un peu de temps. L'Agent SSM ne peut pas traiter les demandes tant que cette actualisation n'a pas été effectuée. Pour accélérer le processus d'actualisation, vous pouvez redémarrer l'Agent SSM ou redémarrer l'instance.

À propos de stratégies pour un profil d'instance Systems Manager

Cette section décrit les stratégies que vous pouvez ajouter à votre profil d'instance EC2 pour Systems Manager. Pour fournir des autorisations de communication entre les instances et l'API Systems Manager, nous vous recommandons de créer des stratégies personnalisées prenant en compte les besoins de votre système et les exigences de sécurité. Toutefois, comme point de départ, vous pouvez utiliser une ou plusieurs des stratégies suivantes pour accorder à Systems Manager l'autorisation d'interagir avec vos instances. La première stratégie, AmazonSSMManagedInstanceCore , permet à une instance d'utiliser les fonctionnalités de base du AWS Systems Manager service. En fonction de votre plan d'opérations, vous pouvez avoir besoin des autorisations représentées dans une ou plusieurs des trois autres stratégies.

Stratégie : AmazonSSMManagedInstanceCore

Autorisations nécessaires.

Cette stratégie gérée AWS permet à une instance d'utiliser les fonctionnalités de base du service Systems Manager.

Stratégie : Une stratégie personnalisée pour l'accès au compartiment S3

Autorisations requises dans l'un des cas suivants :

- Cas 1 : Vous utilisez un point de terminaison de VPC pour connecter de manière privée votre VPC aux services AWS pris en charge et aux services de point de terminaison de VPC alimentés par PrivateLink .

  Agent SSM est un logiciel Amazon qui est installé sur vos instances et effectue des tâches Systems Manager. Cet agent a besoin d'accéder à des compartiments S3 appartenant à Amazon spécifiques. Ces compartiments sont accessibles publiquement.

  Dans un environnement de point de terminaison d'un VPC privé, vous devez toutefois fournir explicitement l'accès à ces compartiments :

  ```
  arn:aws:s3:::patch-baseline-snapshot-region/*
  arn:aws:s3:::aws-ssm-region/*
  ```

  Pour plus d'informations, consultez Étape 6 : (Facultatif) Créer un point de terminaison de VPC (p. 42), À propos des autorisations minimales relatives aux compartiments S3 pour l'Agent SSM (p. 112) et Points de terminaison d'un VPC dans le Amazon VPC Guide de l'utilisateur.
- Cas 2 : Vous prévoyez d'utiliser un compartiment S3 que vous créez dans le cadre de vos Systems Manager opérations .

  Votre profil d'instance EC2 pour Systems Manager doit accorder l'accès à un compartiment S3 que vous possédez pour des tâches comme les suivantes :
  - Pour accéder aux scripts que vous stockez dans le compartiment S3 pour les utiliser dans les commandes que vous exécutez.
  - Pour stocker la sortie complète des commandes Fonctionnalité Exécuter la commande ou des sessions Session Manager.
  - Pour accéder aux listes de correctifs personnalisés à utiliser lors de l'application de correctifs à vos instances.

    Note

    L'enregistrement des données de journal de sortie dans un compartiment S3 est facultatif, mais nous vous recommandons de le configurer au début de votre processus de configuration d'Systems Manager si vous avez décidé de l'utiliser. Pour de plus amples informations, veuillez consulter Créer un compartiment dans le Amazon Simple Storage Service Guide de mise en route.

Stratégie : AmazonSSMDirectoryServiceAccess

Obligatoire uniquement si vous envisagez de joindre une instance EC2 pour Windows Server à un annuaire Microsoft AD.

Cette stratégie gérée AWS permet à Agent SSM d'accéder à AWS Directory Service en votre nom pour les demandes de jonction de domaine par l'instance gérée. Pour plus d'informations, consultez Jonction facile d'une instance EC2 Windows dans le AWS Directory Service Administration Guide.

Stratégie : CloudWatchAgentServerPolicy

Obligatoire uniquement si vous prévoyez d'installer et d'exécuter l'Agent CloudWatch sur vos instances pour lire les métriques et les données de journal sur une instance et de les écrire dans Amazon

CloudWatch. Celles-ci vous permettent de surveiller, d'analyser et de répondre rapidement aux problèmes ou aux modifications apportées à vos ressources AWS.

Votre profil d'instance a besoin de cette stratégie uniquement si vous utilisez des fonctionnalités telles que Amazon EventBridge ou CloudWatch Logs . (Vous pouvez également créer une stratégie plus restrictive qui, par exemple, limite l'accès en écriture à un flux de journaux CloudWatch Logs spécifique.)

> **Note**
>
> L'utilisation EventBridge des CloudWatch Logs fonctionnalités et est facultative, mais nous vous recommandons de les configurer au début de votre processus Systems Manager de configuration si vous avez décidé de les utiliser. Pour plus d'informations, consultez le Guide de l'utilisateur Amazon EventBridge et le Amazon CloudWatch Logs User Guide.

Pour créer un profil d'instance avec des autorisations pour des services Systems Manager supplémentaires, consultez les ressources suivantes.

- Restriction de l'accès aux paramètres Systems Manager à l'aide des stratégies IAM (p. 223)
- Configuration d'Automation (p. 308)
- Vérifier ou créer un profil d'instance IAM avec des autorisations Session Manager (p. 876)
- Configuration d'Fonctionnalité Exécuter la commande (p. 943)

Tâche 1 : (Facultatif) Créer une stratégie personnalisée pour l'accès au compartiment S3

La création d'une stratégie personnalisée pour l'accès Amazon S3 est obligatoire uniquement si vous utilisez le point de terminaison d'un VPC ou votre propre compartiment S3 dans vos opérations Systems Manager.

Pour plus d'informations sur les compartiments S3 gérés AWS auxquels vous fournissez l'accès dans la stratégie ci-dessous, consultez À propos des autorisations minimales relatives aux compartiments S3 pour l'Agent SSM (p. 112).

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation, choisissez Stratégies, puis Créer une stratégie.
3. Choisissez l'onglet JSON et remplacez le texte par défaut avec le texte suivant :

```
{
    "Version": "2012-10-17",
    "Statement": [
        1 {
            "Effect": "Allow",
            "Action": "s3:GetObject",
            "Resource": [
                "arn:aws:s3:::aws-ssm-region/*",
                "arn:aws:s3:::aws-windows-downloads-region/*",
                "arn:aws:s3:::amazon-ssm-region/*",
                "arn:aws:s3:::amazon-ssm-packages-region/*",
                "arn:aws:s3:::region-birdwatcher-prod/*",
                "arn:aws:s3:::aws-ssm-distributor-file-region/*",
                "arn:aws:s3:::aws-ssm-document-attachments-region/*",
                "arn:aws:s3:::patch-baseline-snapshot-region/*"
            ]
        },
        2 {
            "Effect": "Allow",
            "Action": [
```

```
                "s3:GetObject",
                "s3:PutObject",
                "s3:PutObjectAcl", 3
                "s3:GetEncryptionConfiguration" 4
            ],
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*",
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET" 5
            ]
        }
    ]
}
```

[1] Le premier élément `Statement` est obligatoire uniquement si vous utilisez un point de terminaison d'un VPC.

[2] Le deuxième élément `Statement` est obligatoire uniquement si vous utilisez un compartiment S3 que vous avez créé pour l'utiliser dans vos opérations Systems Manager.

[3] L'autorisation de liste de contrôle d'accès `PutObjectAcl` est obligatoire uniquement si vous prévoyez de prendre en charge l'accès entre comptes à des compartiments S3 d'autres comptes.

[4] L'élément `GetEncryptionConfiguration` est obligatoire si votre compartiment S3 est configuré pour utiliser le chiffrement.

[5] Si votre compartiment S3 est configuré pour utiliser le chiffrement, la racine du compartiment S3 (par exemple, arn:aws:s3:::*`DOC-EXAMPLE-BUCKET`*) doit être répertoriée dans la section Ressource. Votre utilisateur, groupe ou rôle IAM doit être configuré avec l'accès au compartiment racine.

4. Si vous utilisez un point de terminaison d'un VPC dans vos opérations, procédez comme suit :

   Dans le premier `Statement` élément, remplacez chaque *`region`* : espace réservé avec l'identifiant de la AWS région dans laquelle cette stratégie sera utilisée. Par exemple, utilisez `us-east-2` pour la région Région USA Est (Ohio). Pour obtenir la liste des pris en charge *`region`* consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services .

   > **Important**
   >
   > Dans cette stratégie, nous vous recommandons d'éviter d'utiliser des caractères génériques (*) à la place des régions spécifiques. Par exemple, utilisez `arn:aws:s3:::aws-ssm-us-east-2/*` et n'utilisez pas `arn:aws:s3:::aws-ssm-*/*` . L'utilisation de caractères génériques pourrait fournir un accès aux compartiments S3 auxquels vous n'avez pas l'intention d'accorder l'accès. Si vous souhaitez utiliser le profil d'instance pour plusieurs régions, nous vous recommandons de répéter le premier élément `Statement` pour chaque région.

   -ou-

   Si vous n'utilisez pas un point de terminaison d'un VPC dans vos opérations, vous pouvez supprimer le premier élément `Statement`.

5. Si vous utilisez votre propre compartiment S3 dans vos opérations Systems Manager, procédez comme suit :

   Dans le deuxième `Statement` élément, remplacez *`DOC-EXAMPLE-BUCKET`* par le nom d'un compartiment S3 dans votre compte. Vous utiliserez ce compartiment pour vos opérations Systems Manager. Il fournit des autorisations pour les objets du compartiment, en utilisant `"arn:aws:s3:::my-bucket-name/*"` comme ressource. Pour plus d'informations sur l'octroi

d'autorisations pour des compartiments ou des objets dans des compartiments , consultez la rubrique Spécification des autorisations dans une stratégie dans l' Amazon Simple Storage Service Manuel du développeur et dans le AWS billet de blog IAM Policies and Bucket Policies and ACLs ! Oh My! (Contrôle de l'accès aux ressources S3)

> **Note**
>
> Si vous utilisez plusieurs compartiments, fournissez l'ARN de chacun d'entre eux. Par exemple, pour les autorisations sur les compartiments :

```
"Resource": [
"arn:aws:s3:::DOC-EXAMPLE-BUCKET1/*",
"arn:aws:s3:::DOC-EXAMPLE-BUCKET2/*"
                ]
```

-ou-

Si vous n'utilisez pas un compartiment S3 vous appartenant dans vos opérations Systems Manager, vous pouvez supprimer le second élément `Statement`.

6. Choisissez Examiner une stratégie.
7. Pour Nom, saisissez un nom pour identifier cette stratégie, par exemple **`SSMInstanceProfileS3Policy`** ou tout autre nom de votre choix.
8. Choisissez Create policy (Créer une stratégie).

Tâche 2 : Ajouter des autorisations à un profil dSystems Manager'instance (console)

Selon que créez un rôle pour votre profil d'instance ou que vous ajoutez les autorisations nécessaires à un rôle existant, utilisez l'une des procédures suivantes.

Pour créer un profil d'instance pour les instances gérées Systems Manager (console)

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le volet de navigation, choisissez Rôles, puis Créer un rôle.
3. Sous Select type of trusted entity (Sélectionner le type d'entité de confiance), choisissez Service AWS.
4. Sous Choisissez immédiatement le service qui utilisera ce rôle, choisissez EC2, puis Suivant : Autorisations.

5. Sur la page Attacher des stratégies d'autorisation, procédez comme suit :
   - Utilisez le champ de recherche pour localiser l' AmazonSSMManagedInstanceCore . Cochez la case en regard de son nom.

La console conserve votre sélection même si vous recherchez d'autres stratégies.

- Si vous avez créé une stratégie de compartiment S3 personnalisée au cours de la procédure précédente, Tâche 1 : (Facultatif) Créer une stratégie personnalisée pour l'accès au compartiment S3 (p. 36), recherchez-la et cochez la case en regard de son nom.
- Si vous prévoyez de joindre des instances à un annuaire Active Directory géré par AWS Directory Service , recherchez AmazonSSMDirectoryServiceAccess et cochez la case en regard de son nom.
- Si vous prévoyez d'utiliser EventBridge ou CloudWatch Logs pour gérer ou surveiller votre instance, recherchez CloudWatchAgentServerPolicy et cochez la case en regard de son nom.

6. Choisissez Suivant : Balises.
7. (Facultatif) Ajoutez une ou plusieurs paires clé-valeur de balise pour organiser, suivre ou contrôler l'accès pour ce rôle, puis choisissez Suivant : Passez en revue.
8. Pour Nom du rôle, entrez un nom pour votre nouveau profil d'instance, par exemple **`SSMInstanceProfile`** ou un autre nom en fonction de vos préférences.

   Note

   Notez le nom de rôle. Choisissez ce rôle lorsque vous créerez de nouvelles instances à gérer à l'aide de Systems Manager.
9. (Facultatif) Dans le champ Description du rôle, saisissez une description pour ce profil d'instance.
10. Sélectionnez Créer un rôle. Le système vous renvoie à la page Rôles.

Pour ajouter des autorisations de profil d'instance pour Systems Manager à un rôle existant (console)

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation, choisissez Rôles, puis choisissez le rôle existant à associer à un profil d'instance pour des opérations Systems Manager.
3. Sous l'onglet Autorisations, choisissez Attach policies (Attacher des stratégies).
4. Sur la page Attach permission policies (Attacher des stratégies d'autorisation), procédez comme suit :
   - Cochez la case en regard de la stratégie AmazonSSMManagedInstanceCore gérée requise.
   - Si vous avez créé une stratégie de compartiment S3 personnalisée, cochez la case en regard de son nom. Pour plus d'informations sur les stratégies de compartiment S3 personnalisés pour un profil d'instance, consultez Tâche 1 : (Facultatif) Créer une stratégie personnalisée pour l'accès au compartiment S3 (p. 36).
   - Si vous prévoyez de joindre des instances à un annuaire Active Directory géré par AWS Directory Service , cochez la case en regard de AmazonSSMDirectoryServiceAccess .
   - Si vous prévoyez d'utiliser EventBridge ou CloudWatch Logs pour gérer ou surveiller votre instance, cochez la case en regard de CloudWatchAgentServerPolicy .
5. Choisissez Attach policy (Attacher la stratégie).

Pour plus d'informations sur la mise à jour d'un rôle en vue d'y inclure une entité de confiance ou d'en limiter davantage l'accès, consultez Modification d'un rôle dans le IAM Guide de l'utilisateur.

Passez au Étape 5 : Attachement d'un profil d'instance IAM à une instance EC2 (p. 40).

Étape 5 : Attachement d'un profil d'instance IAM à une instance EC2

Les procédures de cette rubrique décrivent comment attacher le profil d'instance IAM pour Systems Manager que vous avez créé dans la rubrique précédente, Étape 4 : Créer un profil dIAM'instance pourSystems Manager (p. 34), aux instances EC2. Vous pouvez attacher le profil d'instance à de nouvelles instances EC2 lors de leur lancement ou à des instances EC2 existantes.

Conditions requises concernant l'Agent SSM pour les instances

AWS Systems Manager Agent (Agent SSM) est un logiciel Amazon qui peut être installé et configuré sur une instance EC2, un serveur sur site ou une machine virtuelle (VM). L'Agent SSM permet à Systems Manager de mettre à jour, de gérer et de configurer ces ressources.

Si le type d'Amazon Machine Image (AMI) que vous choisissez dans la première procédure n'est pas livré avec l'Agent SSM préinstallé, vous devez installer l'agent manuellement sur la nouvelle instance pour qu'elle puisse être utilisée avec Systems Manager. Si l'Agent SSM n'est pas installé sur l'instance EC2 existante que vous choisissez dans la deuxième procédure, vous devez installer l'agent manuellement sur l'instance pour qu'elle puisse être utilisée avec Systems Manager.

Agent SSMest installé par défaut sur les éléments suivants AMIs :

- Amazon Linux
- Amazon Linux 2
- AMIs optimisées pour ECS Amazon Linux 2
- Ubuntu Server 16.04, 18.04 et 20.04
- AMIs Windows Server 2008-2012 R2 publiées à partir de novembre 2016
- Windows Server 2016 et 2019

> **Note**
>
> L'Agent SSM n'est pas installé sur toutes les AMIs basées sur Amazon Linux ou Amazon Linux 2. Par exemple, l'Agent SSM n'est pas préinstallé sur les AMIs optimisées pour EKS basées sur Amazon Linux 2.

Pour de plus amples informations sur l'installation manuelle de l'Agent SSM sur d'autres systèmes d'exploitation Linux, veuillez consulter Installation et configuration de l'Agent SSM sur les instances EC2 pour Linux (p. 75).

Conditions requises concernant le certificat TLS pour les instances

Un certificat TLS (Transport Layer Security) doit être installé sur chaque instance gérée que vous utilisez avec Systems Manager. Ces certificats sont utilisés pour chiffrer les appels vers d'autres services AWS. Un certificat TLS est déjà installé sur chaque instance EC2 créée à partir d'une Amazon Machine Image (AMI). Sur les instances créées à partir de AMIs non fournies par Amazon, et sur vos propres serveurs sur site et VMs , vous devez installer le certificat vous-même. Pour plus d'informations, consultez la section Installer un certificat TLS sur des serveurs et des machines virtuelles sur site (p. 54).

Rubriques

- Lancer une instance qui utilise le profil d'instance Systems Manager (console) (p. 41)
- Attacher le profil d'instance Systems Manager à une instance existante (console) (p. 41)

Lancer une instance qui utilise le profil d'instance Systems Manager (console)

Pour lancer une instance qui utilise le profil d'instance Systems Manager (console)

1. Ouvrez la console Amazon EC2 à l'adresse https://console.aws.amazon.com/ec2/.
2. Dans la barre de navigation, en haut de l'écran, sélectionnez la région AWS associée à l'instance.
3. Choisissez Lancer les instances.
4. Sur la page Sélection d'une Amazon Machine Image (AMI), localisez l'AMI pour le type d'instance à créer, puis choisissez Sélectionner.
5. Choisissez le type d'instance à lancer, tel que t2.micro, puis choisissez Suivant : Configurez les détails de l'instance.
6. Sur la page Configurer les détails de l'instance, dans la liste déroulante Rôle IAM, sélectionnez le profil d'instance que vous avez créé via la procédure présentée à l'Étape 4 : Créer un profil dIAM'instance pourSystems Manager (p. 34).
7. Pour les autres options de la page, effectuez les sélections qui répondent à vos exigences concernant l'instance. Pour de plus amples informations, choisissez l'une des options suivantes, en fonction du type de système d'exploitation sélectionné :
   - Linux : Lancement d'une instance à l'aide de l'assistant de lancement d'instance dans laAmazon EC2 Guide de l'utilisateur pour les instances Linux
   - Windows Server : Lancement d'une instance à l'aide de l'assistant de lancement d'instance dans le Amazon EC2 Guide de l'utilisateur pour les instances Windows
8. Exécutez l'assistant.

Si vous créez d'autres instances que vous souhaitez configurer à l'aide de Systems Manager, vous devez spécifier le profil d'instance pour chaque instance.

Attacher le profil d'instance Systems Manager à une instance existante (console)

1. Connectez-vous à AWS Management Console et ouvrez la console Amazon EC2 à l'adresse https://console.aws.amazon.com/ec2/.
2. Dans le panneau de navigation, sous Instances, choisissez Instances.
3. Recherchez et choisissez votre instance EC2 dans la liste.
4. Dans le menu Actions, choisissez Paramètres de l'instance, Attacher/Remplacer le rôle IAM.
5. Pour Rôle IAM, sélectionnez le profil d'instance que vous avez créé via la procédure décrite à l'Étape 4 : Créer un profil dIAM'instance pourSystems Manager (p. 34).
6. Choisissez Apply (Appliquer).

Pour de plus amples informations sur l'attachement de rôles IAM à des instances, veuillez consulter Attachement d'un rôle IAM à une instance dans le Amazon EC2 Guide de l'utilisateur pour les instances Linux.

Passez au Étape 6 : (Facultatif) Créer un point de terminaison de VPC (p. 42).

Étape 6 : (Facultatif) Créer un point de terminaison de VPC

Vous pouvez renforcer la sécurité de vos instances gérées (notamment de celles figurant dans votre environnement hybride) en configurant AWS Systems Manager pour qu'il utilise un point de terminaison de VPC d'interface dans Amazon Virtual Private Cloud (Amazon VPC). Un point de terminaison de VPC d'interface (point de terminaison d'interface) vous permet de vous connecter à des services alimentés par AWSPrivateLink, une technologie qui vous permet d'accéder de façon privée à et à l'aide d'adresses IP privées.Amazon EC2Systems ManagerAPIs PrivateLinklimite l'ensemble du trafic réseau entre vos instances gérées, et Systems Manager au réseau Amazon.Amazon EC2 (Les instances gérées n'ont pas accès à Internet.) En outre, vous n'avez pas besoin d'une passerelle Internet, d'un périphérique NAT ou d'une passerelle réseau privé virtuel.

Il n'est pas obligatoire de configurer PrivateLink, mais c'est recommandé. Pour plus d'informations sur PrivateLink et les points de terminaison d'un VPC, consultez Accès aux AWS services via PrivateLink .

> **Note**
>
> L'alternative à l'utilisation d'un point de terminaison de VPC est l'activation de l'accès Internet sortant sur vos instances gérées. Dans ce cas, les instances gérées doivent également autoriser le trafic sortant HTTPS (port 443) vers les points de terminaison suivants :
>
> - `ssm.`*`region`*`.amazonaws.com`
> - `ssmmessages.`*`region`*`.amazonaws.com`
> - `ec2messages.`*`region`*`.amazonaws.com`
>
> Pour de plus amples informations sur ces points de terminaison, veuillez consulter Référence : ec2messages, ssmmessages et autres appels d'API (p. 1399).

À propos d'Amazon VPC

Amazon Virtual Private Cloud (Amazon VPC) vous permet de définir un réseau virtuel dans votre propre domaine isolé de manière logique dans le cloud AWS, appelé virtual private cloud (VPC). Vous pouvez lancer vos ressources AWS, comme des instances, dans votre VPC. Votre VPC ressemble beaucoup à un réseau traditionnel que vous pourriez exécuter dans votre propre centre de données, et présente l'avantage d'utiliser l'infrastructure évolutive d'AWS. Vous pouvez configurer votre VPC en sélectionnant sa plage d'adresses IP, en créant des sous-réseaux et en configurant des tables de routage, des passerelles réseau et des paramètres de sécurité. Vous pouvez connecter des instances de votre VPC à Internet. Vous pouvez connecter votre VPC à votre propre centre de données d'entreprise, ce qui fait du cloud AWS une extension de votre centre de données. Pour protéger les ressources dans chaque sous-réseau, vous pouvez utiliser plusieurs couches de sécurité, y compris des groupes de sécurité et des listes de contrôle d'accès réseau. Pour de plus amples informations, veuillez consulter Amazon VPC Guide de l'utilisateur.

Rubriques

- Limites et restrictions applicables aux points de terminaison de VPC (p. 42)
- Création des points de terminaison d'un VPC pour Systems Manager (p. 44)

Limites et restrictions applicables aux points de terminaison de VPC

Avant de configurer les points de terminaison d'un VPC pour Systems Manager, tenez compte des restrictions et limitations suivantes.

Plug-in aws:domainJoin

Si vous choisissez de créer des points de terminaison d'un VPC, sachez que les demandes pour joindre une instance Windows Server à un domaine à partir de documents SSM utilisant le plug-in `aws:domainJoin` échoueront. Ce plug-in nécessite l' AWS Directory Service et AWS Directory Service ne prend pas en charge les PrivateLink points de terminaison. La prise en charge de la jonction d'une instance Windows Server à un domaine à partir d'autres méthodes de jonction de domaine dépend des exigences liées à Active Directory (par exemple, il faut s'assurer que les contrôleurs de domaine sont accessibles et détectables à l'aide de DNS ; il existe également d'autres exigences connexes). Vous pouvez utiliser des scripts de données utilisateur Amazon EC2 pour joindre une instance à un domaine.

Demandes inter-régionales

Les points de terminaison d'un VPC ne prennent pas en charge de demandes inter-régionales — pour le moment, c'est pourquoi vous devez veiller à créer votre point de terminaison dans la même région que votre compartiment. Vous pouvez trouver l'emplacement de votre compartiment en utilisant la console Amazon S3 ou avec la commande get-bucket-location. Utilisez un Amazon S3 point de terminaison propre à une région pour accéder à votre compartiment ; par exemple, `DOC-EXAMPLE-BUCKET.s3-us-west-2.amazonaws.com` . Pour plus d'informations sur les points de terminaison spécifiques à une région pour Amazon S3 , consultez Points de terminaison de Amazon S3 service dans le Référence générale d'Amazon Web Services . Si vous utilisez l'AWS CLI pour effectuer des demandes à Amazon S3, configurez votre région par défaut comme la même région que votre compartiment, ou utilisez le paramètre `--region` dans vos demandes.

Connexions d'appairage de VPC

Les points de terminaison de l'interface VPC sont accessibles via des connexions d'appairage de VPC intra-région et inter-région . Pour plus d'informations sur les demandes de connexion d'appairage de VPC pour les points de terminaison de l'interface VPC, consultez Propriétés et limitations des points de terminaison d'interface dans le Guide de l'utilisateur Amazon Virtual Private Cloud.

Les connexions de point de terminaison de passerelle VPC ne peuvent être étendues à l'extérieur d'un VPC. Les ressources de l'autre côté d'une connexion d'appairage de VPC dans votre VPC ne peuvent pas utiliser le point de terminaison de passerelle pour communiquer avec des ressources du service de point de terminaison de passerelle. Pour plus d'informations sur les demandes de connexion d'appairage de VPC pour les points de terminaison de passerelle VPC, consultez Limitations de point de terminaison de passerelle dans le Guide de l'utilisateur Amazon Virtual Private Cloud.

Connexions entrantes

Le groupe de sécurité attaché au point de terminaison d'un VPC doit autoriser les connexions entrantes sur le port 443 à partir du sous-réseau privé de l'instance gérée. Si les connexions entrantes ne sont pas autorisées, l'instance gérée ne peut pas se connecter aux points de terminaison SSM et EC2.

Compartiments Amazon S3

Votre politique de point de terminaison de VPC doit au moins autoriser l'accès aux compartiments Amazon S3 suivants :

- Les compartiments S3 utilisés par Gestionnaire de correctifs pour les opérations de référence de correctifs dans votre AWS région. Ces compartiments contiennent le code qui est récupéré et exécuté sur les instances par le service de référence de correctifs. Chaque AWS région possède ses propres compartiments d'opérations de référence de correctifs à partir desquels le code est récupéré lorsqu'un document de référence de correctifs est exécuté. Si le code ne peut pas être téléchargé, la commande de référence de correctifs échoue.

  > **Note**
  >
  > Si vous utilisez un pare-feu sur site et que vous prévoyez d'utiliser Gestionnaire de correctifs , ce pare-feu doit également autoriser l'accès au point de terminaison de référence de correctif indiqué ci-dessous.

Pour fournir l'accès aux compartiments dans votre région AWS, incluez l'autorisation suivante dans votre stratégie de point de terminaison :

> **Note**
>
> Dans la région Région Moyen-Orient (Bahrain) (me-south-1) uniquement, ces compartiments utilisent une convention de dénomination différente. Pour cette région AWS uniquement, utilisez plutôt les deux compartiments suivants.
>
> - `patch-baseline-snapshot-me-south-1-uduvl7q8`
> - `aws-patch-manager-me-south-1-a53fc9dce`

```
arn:aws:s3:::patch-baseline-snapshot-region/*
arn:aws:s3:::aws-ssm-region/*
```

`région` représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de *`region`* prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

Exemple :

```
arn:aws:s3:::patch-baseline-snapshot-us-east-2/*
arn:aws:s3:::aws-ssm-us-east-2/*
```

- Les compartiments S3 répertoriés dans À propos des autorisations minimales relatives aux compartiments S3 pour l'Agent SSM (p. 112).

Amazon CloudWatch Logs

Si vous n'autorisez pas vos instances à accéder à Internet, vous devez créer un point de terminaison de VPC pour CloudWatch Logs afin d'utiliser des fonctions qui envoient des journaux à CloudWatch Logs . Pour plus d'informations sur la création d'un point de terminaison pour CloudWatch Logs , consultez Création d'un point de terminaison de VPC pour CloudWatch Logs dans le Amazon CloudWatch Logs User Guide .

DNS dans un environnement hybride

Pour plus d'informations sur la configuration de DNS pour utiliser des PrivateLink points de terminaison dans des environnements hybrides, consultez DNS privé. Si vous souhaitez utiliser votre propre DNS, vous pouvez utiliser Route 53 Resolver. Pour plus d'informations, consultez Résolution des requêtes DNS entre VPCs et votre réseau dans le Amazon Route 53 Manuel du développeur .

Création des points de terminaison d'un VPC pour Systems Manager

Utilisez les informations suivantes pour créer une interface VPC et des points de terminaison de passerelle pour Systems Manager. Cette rubrique renvoie aux procédures du Amazon VPC Guide de l'utilisateur.

Pour créer les points de terminaison d'un VPC pour Systems Manager

Dans la première étape ci-dessous, vous créez trois points de terminaison d'interface obligatoires et un optionnel pour Systems Manager. Les trois points de terminaison sont requis pour que Systems Manager fonctionne dans un VPC. Le quatrième, `com.amazonaws.`*`region`*`.ssmmessages`, est uniquement obligatoire si vous utilisez des capacités Session Manager.

Dans la deuxième étape, vous créez le point de terminaison de passerelle requis pour que Systems Manager accède à Amazon S3.

> **Note**
>
> *région* représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de *region* prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

1. Suivez les étapes dans Création d'un point de terminaison d'interface pour créer les points de terminaison suivants :
   - com.amazonaws.***region***.ssm : Point de terminaison du Systems Manager service.
   - com.amazonaws.***region***.ec2messages : Systems Manager utilise ce point de terminaison pour effectuer des appels d' Agent SSM au Systems Manager service .
   - com.amazonaws.***region***.ec2 : Si vous utilisez Systems Manager pour créer des instantanés activés pour VSS, vous devez vous assurer que vous disposez d'un point de terminaison pour le service EC2. Sans le point de terminaison EC2 défini, un appel pour énumérer les volumes EBS attachés échoue, ce qui entraîne l'échec de la commande Systems Manager.
   - com.amazonaws.***region***.ssmmessages : Ce point de terminaison est obligatoire uniquement si vous vous connectez à vos instances via un canal de données sécurisé à l'aide d' Session Manager . Pour de plus amples informations, veuillez consulter AWS Systems Manager Session Manager (p. 869) et Référence : ec2messages, ssmmessages et autres appels d'API (p. 1399).
2. Suivez les étapes dans Création d'un point de terminaison de passerelle pour créer le point de terminaison de passerelle suivant pour Amazon S3.
   - com.amazonaws.***region***.s3 : Systems Manager utilise ce point de terminaison pour mettre à jour Agent SSM et pour des tâches telles que le chargement des journaux de sortie que vous choisissez de stocker dans des compartiments S3, la récupération de scripts ou d'autres fichiers que vous stockez dans des compartiments , etc.

Passez au Étape 7 : (Facultatif) Créer des rôles de service Systems Manager (p. 45).

Étape 7 : (Facultatif) Créer des rôles de service Systems Manager

Cette rubrique explique la différence entre un rôle de service et un rôle lié à un service pour Systems Manager. Elle explique également dans quels cas vous devez utiliser l'un ou l'autre de ces types de rôle.

Rôle de service : Un rôle de service est un AWS Identity and Access Management (IAM) qui accorde des autorisations à un AWS service afin qu'il puisse accéder aux AWS ressources . Seuls quelques scénarios Systems Manager nécessitent un rôle de service. Lorsque vous créez un rôle de service pour Systems Manager, vous choisissez les autorisations à accorder pour qu'il puisse accéder aux autres ressources AWS ou interagir avec ces dernières.

Rôle lié à un service : Un rôle lié à un service est prédéfini par Systems Manager et inclut toutes les autorisations nécessaires au service pour appeler d'autres AWS services en votre nom.

Actuellement, le rôle lié à un service Systems Manager peut être utilisé dans les cas suivants :

- La fonctionnalité d'inventaire Systems Manager utilise le rôle lié à un service pour collecter les métadonnées d'inventaire à partir des balises et des groupes de ressources.

- La fonctionnalité Fenêtres de maintenance peut utiliser le rôle lié à un service dans certaines situations. D'autres situations nécessitent un rôle de service personnalisé que vous créez, comme décrit ci-dessous.

Pour de plus amples informations sur le rôle lié à un service, veuillez consulter Utilisation des rôles liés à un service pour Systems Manager (p. 1312).

Création d'un rôle de service

Vous pouvez créer les rôles de service suivants dans le cadre de l'installation de Systems Manager ou vous pouvez les créer plus tard.

Rôle de service pour Automation

Auparavant, l'automatisation nécessitait que vous spécifiiez un rôle de service afin que ce service soit autorisé à réaliser des actions en votre nom. Automation ne nécessite plus ce rôle car le service opère désormais à l'aide du contexte de l'utilisateur ayant invoqué l'exécution.

Néanmoins, les situations suivantes nécessitent tout de même que vous spécifiiez un rôle du service pour l'automatisation :

- Lorsque vous souhaitez restreindre les privilèges d'un utilisateur sur une ressource tout en souhaitant que l'utilisateur puisse exécuter un flux de travail d'automatisation nécessitant des privilèges supérieurs. Dans ce scénario, vous pouvez créer un rôle de service avec des privilèges supérieurs et autoriser l'utilisateur à exécuter le flux de travail.
- Les opérations qui seront probablement exécutées pendant plus de 12 heures nécessitent un rôle de service.

Si vous avez besoin de créer un rôle de service et un rôle de profil d'instance pour l'automatisation, vous pouvez utiliser l'une des méthodes suivantes.

- Méthode 1 : Utiliser AWS CloudFormation pour configurer un rôle de service pour Automation (p. 309)
- Méthode 2 : Utiliser IAM pour configurer des rôles pour Automation (p. 310)

Rôle de service pour les tâches de la fenêtre de maintenance

Pour pouvoir exécuter des tâches sur vos instances gérées, le service Fenêtres de maintenance doit avoir l'autorisation d'accéder à ces ressources. Cette autorisation peut être accordée à l'aide d'un rôle lié à un service pour Systems Manager ou d'un rôle de service personnalisé que vous créez.

Vous créez un rôle de service personnalisé dans les cas suivants :

- Si vous souhaitez utiliser un ensemble d'autorisations plus restrictif que celles fournies par le rôle lié à un service.
- Si vous souhaitez utiliser un ensemble d'autorisations plus permissif ou étendu que celles fournies par le rôle lié à un service. Par exemple, certaines actions dans les documents Automation nécessitent des autorisations pour des actions dans d'autres services AWS.

Pour de plus amples informations, veuillez consulter les rubriques suivantes dans la section Fenêtres de maintenance du présent guide de l'utilisateur :

- Dois-je utiliser un rôle lié à un service ou un rôle de service personnalisé pour exécuter des tâches de fenêtre de maintenance ? (p. 691)
- (Facultatif) Créer un rôle de service personnalisé pour les Fenêtres de maintenance (console) (p. 692).

Rôle de service pour les notifications Amazon Simple Notification Service

Amazon Simple Notification Service (Amazon SNS) est un service web qui coordonne et gère la mise à disposition ou l'envoi de messages à des clients ou à des points de terminaison abonnés. Dans Systems Manager, vous pouvez configurer Amazon SNS pour envoyer des notifications sur le statut des commandes que vous envoyez à l'aide de la fonction Fonctionnalité Exécuter la commande ou sur le statut des tâches exécutées dans les fenêtres de maintenance.

Vous créez un rôle de service pour Amazon SNS dans le cadre du processus de configuration du service pour une utilisation avec Systems Manager. Une fois que vous avez terminé cette configuration, vous pouvez choisir de recevoir des notifications pour des commandes Fonctionnalité Exécuter la commande ou des tâches de fenêtres de maintenance spécifiques au moment de la création de chacune d'elles.

Pour plus d'informations, consultez la section Surveillance des changements de statut Systems Manager à l'aide de notifications Amazon SNS (p. 1344).

Rôle de service pour un environnement hybride Systems Manager

Si vous prévoyez d'utiliser Systems Manager pour gérer des serveurs et des machines virtuelles (VM) sur site dans ce que l'on appelle un environnement hybride, vous devez créer un rôle IAM pour que ces ressources puissent communiquer avec le service Systems Manager.

Pour plus d'informations, consultez la section Créer un rôle de service IAM pour un environnement hybride (p. 50).

Passez au Étape 8 : (Facultatif) Configuration des intégrations avec d'autres AWS services (p. 47).

Étape 8 : (Facultatif) Configuration des intégrations avec d'autres AWS services

AWS Systems Manager s'intègre à un certain nombre d'autres services AWS. Dans la plupart des cas, vous configurez une intégration après avoir décidé d'intégrer le service dans vos opérations Systems Manager. Exemple :

- Référencement des secrets AWS Secrets Manager à partir des paramètres Parameter Store (p. 134)
- Utilisation des InSpec profils Chef avec Systems Manager la conformité (p. 145)

Vous pouvez utiliser certains services AWS immédiatement pour compiler les données de journal en vue des tâches ultérieures de dépannage et d'analyse. Vous pouvez également utiliser les services AWS pour surveiller les modifications apportées à votre environnement Systems Manager et y répondre rapidement. Par conséquent, nous vous recommandons de configurer les ressources suivantes dans le cadre de la procédure de configuration initiale de Systems Manager.

Amazon EventBridge et Amazon Simple Notification Service – EventBridge vous permet de détecter à quel moment les changements ont lieu pour les ressources AWS que vous spécifiez. Vous pouvez configurer EventBridge pour journaliser les changements de statut d'exécution des commandes que les utilisateurs de votre compte envoient à l'aide de Systems Manager. Vous pouvez créer une règle pour détecter les moments auxquels un utilisateur de votre organisation démarre ou arrête une session dans Session Manager. Vous pouvez également configurer un EventBridge événement pour déclencher d'autres actions dans votre AWS environnement. Pour de plus amples informations, consultez les rubriques suivantes :

- Comprendre les états des commandes (p. 965)
- Configuration de EventBridge pour Systems Manager événements (p. 1341)
- Surveillance de l'activité de la session avec Amazon EventBridge (console) (p. 933)

Amazon Simple Storage Service (Amazon S3)

La sortie de commande Fonctionnalité Exécuter la commande dan la console Systems Manager est tronquée après 2500 caractères. Pour accéder aux journaux de sortie des commandes complets, vous pouvez stocker la Systems Manager dans un compartiment Amazon Simple Storage Service (Amazon S3), puis utiliser cette sortie ultérieurement à des fins d'audit ou de dépannage. Vous spécifiez si vous souhaitez enregistrer la sortie de commande dans un compartiment S3 à chaque exécution d'une commande. Vous pouvez également créer un préfixe de clé Amazon S3 (un sous-dossier) afin de vous aider à organiser la sortie de journal. Pour de plus amples informations, veuillez consulter Créer un compartiment dans le Amazon Simple Storage Service Guide de mise en route.

Amazon CloudWatch Logs (CloudWatch Logs)

Au lieu de stocker la sortie de commande dans un compartiment S3, vous pouvez l'envoyer dans un groupe de journaux Amazon CloudWatch Logs. Si vous spécifiez CloudWatch Logs en tant que cible de sortie, Fonctionnalité Exécuter la commande envoie périodiquement toutes les sorties de commande et tous les journaux d'erreur vers CloudWatch Logs. Vous pouvez surveiller les journaux de sortie quasiment en temps réel, rechercher des phrases, valeurs ou modèles spécifiques, et créer des alarmes sur la base de la recherche. Pour plus d'informations, consultez la section Configuration d'Amazon CloudWatch Logs pour Fonctionnalité Exécuter la commande (p. 1338).

Configuration d'AWS Systems Manager pour les environnements hybrides

Cette section décrit les tâches de configuration effectuées par les administrateurs système et de compte pour un environnement hybride. Un environnement hybride inclut des serveurs et des machines virtuelles (VM) sur site qui ont été configurés pour une utilisation avec Systems Manager , y compris VMs dans d'autres environnements cloud. Une fois ces étapes terminées, les utilisateurs qui se sont vus accorder des autorisations par l'administrateur de compte AWS peuvent utiliser AWS Systems Manager pour configurer et gérer les serveurs et les machines virtuelles sur site de leur organisation.

Si vous prévoyez d'utiliser Systems Manager pour gérer des instances Amazon Elastic Compute Cloud (EC2), ou d'utiliser des instances EC2 et vos propres ressources dans un environnement hybride, commencez par suivre les étapes de la section Configuration d'AWS Systems Manager (p. 28).

La configuration de votre environnement hybride pour Systems Manager vous permet de :

- Créer une procédure cohérente et sécurisée pour gérer à distance vos charges de travail hybrides depuis un emplacement unique, à l'aide des mêmes outils ou des mêmes scripts.
- Centralisez le contrôle d'accès pour les actions qui peuvent être effectuées sur vos serveurs et à l'VMsaide d' AWS Identity and Access Management (IAM).
- Centralisez l'audit et obtenez une vue d'ensemble des actions effectuées sur vos serveurs et VMs en enregistrant toutes les actions dans AWS CloudTrail .

  Pour de plus amples informations sur l'utilisation de CloudTrail pour surveiller les actions Systems Manager, veuillez consulter Journalisation des appels d'API AWS Systems Manager avec AWS CloudTrail (p. 1336).
- Centraliser la surveillance en configurant EventBridge et Amazon SNS de sorte à envoyer des notifications sur la réussite de l'exécution de service.

  Pour de plus amples informations sur l'utilisation de EventBridge pour surveiller les événements Systems Manager, veuillez consulter Surveillance des événements Systems Manager avec Amazon EventBridge (p. 1340).

À propos des instances gérées

Une fois que vous avez terminé de configurer vos serveurs et VMs pour Systems Manager comme décrit dans cette section, vos machines hybrides sont répertoriées dans le AWS Management Console et décrites comme des instances gérées. Les instances EC2 configurées pour Systems Manager sont également décrites comme des instances gérées. Dans la console, toutefois, l' IDs de vos instances hybrides se distingue des instances EC2 avec le préfixe « mi- ». IDsLes instances EC2 utilisent le préfixe « i- ».

Pour plus d'informations, consultez la section Instances gérées AWS Systems Manager (p. 851).

À propos des niveaux d'instances

AWS Systems Managerpropose un niveau d'instances standard et un niveau d'instances avancées pour les serveurs et VMs dans votre environnement hybride. Le niveau d'instances standard vous permet d'enregistrer 1 000 serveurs sur site au maximum ou VMs par AWS compte et par AWS région. Si vous devez enregistrer plus de 1 000 serveurs sur site ou VMs dans un seul compte et une seule région, utilisez le niveau des instances avancées. Les instances avancées vous permettent également de vous connecter

à vos machines hybrides en utilisant AWS Systems Manager Session Manager. Session Manager fournit l'accès au shell interactif à vos instances.

Pour plus d'informations, consultez la section Configuration des niveaux d'instance (p. 853).

Rubriques

- Étape 1 : Effectuer les étapes générales Systems Manager de configuration (p. 50)
- Étape 2 : Créer un rôle de service IAM pour un environnement hybride (p. 50)
- Étape 3 : Installer un certificat TLS sur les serveurs sur site etVMs (p. 54)
- Étape 4 : Créer une activation d'instance gérée pour un environnement hybride (p. 55)
- Étape 5 : Installer Agent SSM pour un environnement hybride (Linux) (p. 58)
- Étape 6 : Installer Agent SSM pour un environnement hybride (Windows) (p. 62)

Étape 1 : Effectuer les étapes générales Systems Manager de configuration

Si ce n'est pas déjà fait, effectuez les étapes de configuration générales suivantes pour Systems Manager dans la section Configuration d'AWS Systems Manager (p. 28) du présent Guide de l'utilisateur. Les autres étapes de cette section ne sont obligatoires que si vous envisagez de gérer des instances EC2.

- Inscrivez-vous à AWS (p. 28)
- Créer un utilisateur IAM administrateur pour AWS (p. 29)
- Créer des utilisateurs et des groupes IAM non administrateurs pour Systems Manager (p. 30)
- (Facultatif) Créer un point de terminaison de VPC (p. 42)
- (Facultatif) Créer des rôles de service Systems Manager (p. 45)
- (Facultatif) Configurer les intégrations avec d'autres services AWS (p. 47)

Après vous être assuré que ces étapes ont bien été réalisées, passez à l'Étape 2 : Créer un rôle de service IAM pour un environnement hybride (p. 50).

Étape 2 : Créer un rôle de service IAM pour un environnement hybride

Les serveurs et machines virtuelles appartenant à un environnement hybride nécessitent un rôle IAM pour communiquer avec le service Systems Manager. Ce rôle accorde le droit `AssumeRole` au service Systems Manager. Vous devez uniquement créer le rôle de service pour un environnement hybride une fois pour chaque compte AWS.

> **Note**
>
> Les utilisateurs de votre entreprise ou organisation qui utilisent Systems Manager sur vos machines hybrides doivent recevoir l'autorisation dans IAM d'appeler l'API SSM. Pour plus d'informations, consultez la section Créer des utilisateurs et des groupes IAM non administrateurs pour Systems Manager (p. 30).

Exigence pour les stratégies de compartiment S3

Dans les cas suivants, vous devez créer une stratégie d'autorisation IAM personnalisée pour les compartiments S3 avant de terminer cette procédure :

- Cas 1 : Vous utilisez un point de terminaison de VPC pour connecter de manière privée votre VPC aux services AWS pris en charge et aux services de point de terminaison de VPC alimentés par PrivateLink .
- Cas 2 : Vous prévoyez d'utiliser un compartiment S3 que vous créez dans le cadre de vos Systems Manager opérations, par exemple pour stocker la sortie des Fonctionnalité Exécuter la commande commandes ou Session Manager des sessions dans un compartiment S3. Avant de continuer, suivez les étapes de Créer une stratégie de compartiment S3 personnalisée pour un profil d'instance (p. 36). Les informations sur les stratégies de compartiment S3 de cette rubrique s'appliquent également à votre rôle de service.

  Note

  Si vous utilisez un pare-feu sur site et que vous prévoyez d'utiliser Gestionnaire de correctifs , ce pare-feu doit également autoriser l'accès au point de terminaison de la référence de correctifs`arn:aws:s3:::patch-baseline-snapshot-`*`région`*`/*`.
  `région` représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de *`region`* prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

AWS CLI

Pour créer un rôle de service IAM pour un environnement hybride (AWS CLI)

1. Créez un fichier texte avec un nom tel que `SSMService-Trust.json` avec la stratégie d'approbation suivante. Assurez-vous d'enregistrer le fichier avec l'extension de fichier `.json`.

```
{
            "Version": "2012-10-17",
            "Statement": {
                "Effect": "Allow",
                "Principal": {"Service": "ssm.amazonaws.com"},
                "Action": "sts:AssumeRole"
            }
            }
```

2. Utilisez la commande create-role pour créer le rôle de service. Cet exemple crée un rôle nommé `SSMServiceRole` . Vous pouvez choisir un autre nom si vous préférez.

   Linux

```
aws iam create-role \
    --role-name SSMServiceRole \
    --assume-role-policy-document file://SSMService-Trust.json
```

   Windows

```
aws iam create-role ^
    --role-name SSMServiceRole ^
    --assume-role-policy-document file://SSMService-Trust.json
```

3. Utilisez attach-role-policy comme suit pour permettre au rôle de service que vous venez de créer de créer un jeton de session. Ce jeton de session autorise votre instance gérée à exécuter des commandes à l'aide de Systems Manager.

   Note

   Les stratégies que vous ajoutez pour un profil de service pour des instances gérées dans un environnement hybride sont les mêmes stratégies que celles utilisées pour créer un profil d'instance pour des instances EC2. Pour plus d'informations sur les stratégies AWS

utilisées dans les commandes suivantes, consultez Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34).

(Obligatoire) Exécutez la commande suivante permettre à une instance gérée d'utiliser les fonctionnalités de base du service AWS Systems Manager.

Linux

```
aws iam attach-role-policy \
    --role-name SSMServiceRole \
    --policy-arn arn:aws:iam::aws:policy/AmazonSSMManagedInstanceCore
```

Windows

```
aws iam attach-role-policy ^
    --role-name SSMServiceRole ^
    --policy-arn arn:aws:iam::aws:policy/AmazonSSMManagedInstanceCore
```

Si vous avez créé une stratégie de compartiment S3 personnalisée pour votre rôle de service, exécutez la commande suivante pour permettre à Agent SSM d'accéder aux compartiments que vous avez spécifiés dans la stratégie. Remplacez *account-id* et *my-bucket-policy-name* avec votre ID de AWS compte et le nom de votre compartiment.

Linux

```
aws iam attach-role-policy \
    --role-name SSMServiceRole \
    --policy-arn arn:aws:iam::account-id:policy/my-bucket-policy-name
```

Windows

```
aws iam attach-role-policy ^
    --role-name SSMServiceRole ^
    --policy-arn arn:aws:iam::account-id:policy/my-bucket-policy-name
```

(Facultatif) Exécutez la commande suivante pour autoriser Agent SSM à accéder à AWS Directory Service en votre nom pour les demandes de jonction de domaine par l'instance gérée. Votre profil d'instance a besoin de cette stratégie uniquement si vous joignez vos instances à un annuaire Microsoft AD.

Linux

```
aws iam attach-role-policy \
    --role-name SSMServiceRole \
    --policy-arn arn:aws:iam::aws:policy/AmazonSSMDirectoryServiceAccess
```

Windows

```
aws iam attach-role-policy ^
    --role-name SSMServiceRole ^
    --policy-arn arn:aws:iam::aws:policy/AmazonSSMDirectoryServiceAccess
```

(Facultatif) Exécutez la commande suivante pour permettre à Agent CloudWatch de s'exécuter sur vos instances gérées. Cette commande permet de lire les informations sur une instance et de les écrire dans CloudWatch. Votre profil de service a besoin de cette stratégie uniquement si vous utilisez des services tels que Amazon EventBridge ou Amazon CloudWatch Logs .

```
aws iam attach-role-policy \
    --role-name SSMServiceRole \
    --policy-arn arn:aws:iam::aws:policy/CloudWatchAgentServerPolicy
```

Outils pour PowerShell

Pour créer un rôle de service IAM pour un environnement hybride (Outils AWS pour Windows PowerShell)

1. Créez un fichier texte avec un nom tel que `SSMService-Trust.json` avec la stratégie d'approbation suivante. Assurez-vous d'enregistrer le fichier avec l'extension de fichier `.json`.

```
{
            "Version": "2012-10-17",
            "Statement": {
                "Effect": "Allow",
                "Principal": {"Service": "ssm.amazonaws.com"},
                "Action": "sts:AssumeRole"
            }
            }
```

2. Utilisez New-IAMRole comme suit pour créer un rôle de service. Cet exemple crée un rôle nommé `SSMServiceRole` . Vous pouvez choisir un autre nom si vous préférez.

```
New-IAMRole `
    -RoleName SSMServiceRole `
    -AssumeRolePolicyDocument (Get-Content -raw SSMService-Trust.json)
```

3. Utilisez Register-IAMRolePolicy comme suit pour permettre au rôle de service que vous avez créé de créer un jeton de session. Ce jeton de session autorise votre instance gérée à exécuter des commandes à l'aide de Systems Manager.

   Note

   Les stratégies que vous ajoutez pour un profil de service pour des instances gérées dans un environnement hybride sont les mêmes stratégies que celles utilisées pour créer un profil d'instance pour des instances EC2. Pour plus d'informations sur les stratégies AWS utilisées dans les commandes suivantes, consultez Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34).

   (Obligatoire) Exécutez la commande suivante permettre à une instance gérée d'utiliser les fonctionnalités de base du service AWS Systems Manager.

```
Register-IAMRolePolicy `
    -RoleName SSMServiceRole `
    -PolicyArn arn:aws:iam::aws:policy/AmazonSSMManagedInstanceCore
```

   Si vous avez créé une stratégie de compartiment S3 personnalisée pour votre rôle de service, exécutez la commande suivante pour permettre à Agent SSM d'accéder aux compartiments que vous avez spécifiés dans la stratégie. Remplacez *account-id* et *my-bucket-policy-name* avec votre ID de AWS compte et le nom de votre compartiment.

```
Register-IAMRolePolicy `
    -RoleName SSMServiceRole `
    -PolicyArn arn:aws:iam::account-id:policy/my-bucket-policy-name
```

(Facultatif) Exécutez la commande suivante pour autoriser Agent SSM à accéder à AWS Directory Service en votre nom pour les demandes de jonction de domaine par l'instance gérée. Votre profil d'instance a besoin de cette stratégie uniquement si vous joignez vos instances à un annuaire Microsoft AD.

```
Register-IAMRolePolicy `
    -RoleName SSMServiceRole `
    -PolicyArn arn:aws:iam::aws:policy/AmazonSSMDirectoryServiceAccess
```

(Facultatif) Exécutez la commande suivante pour permettre à Agent CloudWatch de s'exécuter sur vos instances gérées. Cette commande permet de lire les informations sur une instance et de les écrire dans CloudWatch. Votre profil de service a besoin de cette stratégie uniquement si vous utilisez des services tels que Amazon EventBridge ou Amazon CloudWatch Logs .

```
Register-IAMRolePolicy `
    -RoleName SSMServiceRole `
    -PolicyArn arn:aws:iam::aws:policy/CloudWatchAgentServerPolicy
```

Note

Les utilisateurs de votre entreprise ou organisation qui utiliseront Systems Manager sur vos machines hybrides doivent être autorisés dans IAM à appeler l'API SSM. Pour plus d'informations, consultez la section Créer des utilisateurs et attribuer des autorisations (p. 32).

Passez au Étape 3 : Installer un certificat TLS sur les serveurs sur site etVMs (p. 54).

Étape 3 : Installer un certificat TLS sur les serveurs sur site etVMs

Un protocole TLS (Transport Layer Security) doit être installé sur chaque instance gérée que vous utilisez avec des services Systems Manager. Les services AWS utilisent ces certificats pour chiffrer les appels vers les autres services AWS.

Un certificat TLS est déjà installé par défaut sur chaque instance EC2 créée à partir d'une AMI (Amazon Machine Image). La plupart des systèmes d'exploitation modernes incluent le certificat TLS requis par Amazon Trust Services CAs dans leur magasin d'approbations.

Si vous constatez que les certificats CA Amazon Trust Services requis ne sont pas installés sur vos systèmes d'exploitation de base, sur les instances créées à partir d' AMIs qui ne sont pas fournies par Amazon, ou sur vos propres serveurs sur site et VMs , vous devez installer et activer un certificat à partir d'Amazon Trust Services, ou utiliser AWS Certificate Manager (ACM) pour créer et gérer des certificats pour un service intégré pris en charge.

Chacune de vos instances gérées doit avoir l'un des certificats TLS (Transport Layer Security) suivants installé.

- Amazon Root CA 1
- Starfield Services Root Certificate Authority - G2

- Starfield Class 2 Certificate Authority

Pour de plus amples informations sur l'utilisation de ACM, veuillez consulter le AWS Certificate Manager Guide de l'utilisateur.

Si les certificats figurant dans votre environnement informatique sont gérés par un objet de stratégie de groupe (GPO), vous pouvez avoir besoin de configurer une stratégie de groupe pour inclure l'un de ces certificats.

Pour plus d'informations sur les certificats Amazon Root et Starfield, consultez l'article de blog How to Prepare for AWS's Move to Its Own Certificate Authority (Comment se préparer pour le déplacement de AWS vers sa propre autorité de certification).

Passez au Étape 4 : Créer une activation d'instance gérée pour un environnement hybride (p. 55).

Étape 4 : Créer une activation d'instance gérée pour un environnement hybride

Pour configurer les serveurs et les machines virtuelles de votre environnement hybride en tant qu'instances gérées, vous devez créer une activation d'instance gérée. Une fois que l'activation a abouti, vous recevez immédiatement un code d'activation et un ID d'activation. Vous spécifiez cette combinaison code/ID lorsque vous installez Agent SSM sur des serveurs et VMs dans votre environnement hybride. La combinaison code/ID fournit un accès sécurisé au service Systems Manager à partir de vos instances gérées.

> **Important**
>
> Systems Manager renvoie immédiatement le code d'activation et l'ID à la console ou la fenêtre de commande, selon la méthode de création de l'activation. Copiez ces informations et stockez-les en lieu sûr. Si vous quittez la console ou fermez la fenêtre de commande, vous risquez de perdre ces informations. Si vous les perdez, vous devrez recréer une activation.

À propos des expirations d'activation

Une expiration d'activation est une fenêtre de temps pendant laquelle vous pouvez enregistrer des machines sur site auprès de Systems Manager. Une activation expirée n'a aucun impact sur les serveurs et machines virtuelles que vous avez préalablement enregistrés avec Systems Manager. Si une activation expire, vous ne pouvez pas enregistrer d'autres serveurs ou VMs avec en utilisant cette activation spécifique.Systems Manager Il vous suffit d'en créer une.

Chaque serveur et chaque machine virtuelle sur site que vous avez précédemment enregistré(e) reste enregistré(e) comme une instance gérée de Systems Manager tant que vous n'aurez pas annulé son enregistrement de manière explicite. Vous pouvez annuler l'enregistrement d'une instance gérée sur la page Instances gérées de la console Systems Manager à l'aide de la commande de l'AWS CLI deregister-managed-instance, ou à l'aide de l'action d'API DeregisterManagedInstance.

À propos des balises d'activation

Si vous créez une activation à l'aide de l'AWS CLI ou de d'Outils AWS pour Windows PowerShell, vous pouvez spécifier des balises. Les balises sont des métadonnées facultatives que vous affectez à une ressource. Les balises vous permettent de classer une ressource de différentes façons, par exemple, par objectif, par propriétaire ou par environnement. Voici un exemple de commande de l'AWS CLI à exécuter sur une machine Linux locale qui inclut des balises.

```
aws ssm create-activation \
  --default-instance-name MyWebServers \
  --description "Activation for Finance department webservers" \
```

```
--iam-role service-role/AmazonEC2RunCommandRoleForManagedInstances \
--registration-limit 10 \
--region us-east-2 \
--tags "Key=Department,Value=Finance"
```

Si vous spécifiez des balises lorsque vous créez une activation, ces balises sont automatiquement affectées à vos serveurs sur site et VMs lorsque vous les activez.

Vous ne pouvez pas ajouter ou supprimer des balises dans une activation existante. Si vous ne souhaitez pas qu' attribue automatiquement des balises à vos serveurs sur site et utilise une activation, vous pouvez leur ajouter des balises ultérieurement.VMs Plus spécifiquement, vous pouvez baliser vos serveurs sur site et VMs une fois qu'ils se sont connectés à Systems Manager pour la première fois. Une fois qu'ils se sont connectés, un ID d'instance gérée leur est affecté et ils sont répertoriés dans la console Systems Manager avec un ID dont le préfixe est « mi- ». Pour de plus amples informations sur la façon d'ajouter des balises à vos instances gérées sans utiliser le processus d'activation, veuillez consulter Balisage des instances gérées (p. 1369).

> **Note**
>
> Vous ne pouvez pas attribuer des balises à une activation si vous la créez à l'aide de la console Systems Manager. Vous devez la créer à l'aide de l'AWS CLI ou de Outils pour Windows PowerShell.

Si vous ne souhaitez plus gérer un serveur local ou une machine virtuelle (VM) à l'aide de Systems Manager, vous pouvez annuler son enregistrement. Pour plus d'informations, consultez la section Annuler l'enregistrement des instances gérées dans un environnement hybride (p. 864).

Rubriques

- Création d'une activation (console) (p. 56)
- Création d'une activation d'instance gérée (ligne de commande) (p. 57)

Création d'une activation (console)

Pour créer une activation d'instance gérée

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Hybrid Activations (Activations hybrides).

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Hybrid Activations (Activations hybrides).
3. Choisissez Créer une activation.
4. (Facultatif) Dans le champ Description de l'activation, saisissez une description pour cette activation. La description est facultative, mais nous vous recommandons d'en saisir une si vous prévoyez d'activer un grand nombre de serveurs et VMs .
5. Dans le champ Limite d'instance, spécifiez le nombre total de serveurs sur site ou VMs que vous souhaitez enregistrer auprès d'AWS dans le cadre de cette activation. La valeur par défaut est 1.
6. Dans la section Nom du IAM rôle, choisissez une option de rôle de service qui permet VMs à vos serveurs et de communiquer avec AWS Systems Manager dans le cloud :

   a. Choisissez Utiliser le rôle d'exécution de commande par défaut créé par le système pour utiliser un rôle et une stratégie gérée créés par AWS.
   b. Choisissez Sélectionner un rôle IAM personnalisé existant ayant les autorisations requises pour utiliser le rôle personnalisé facultatif que vous avez créé précédemment.

7. Dans le champ Date d'expiration de l'activation, indiquez une date d'expiration pour l'activation. La valeur par défaut est de 24 heures.

   Note

   Si vous souhaitez enregistrer des instances gérées supplémentaires après la date d'expiration, vous devez créer une nouvelle activation. La date d'expiration n'a aucun impact sur les instances enregistrées et en cours d'exécution.
8. (Facultatif) Dans le champ Nom de l'instance par défaut, indiquez un nom.
9. Choisissez Créer une activation. Systems Manager renvoie immédiatement le code d'activation et l'ID à la console.

Création d'une activation d'instance gérée (ligne de commande)

La procédure suivante décrit comment utiliser l'AWS CLI (sous Linux ou Windows) ou Outils AWS pour PowerShell pour créer une activation d'instance gérée.

Pour créer une activation

1. Installez et configurez l'AWS CLI ou Outils AWS pour PowerShell si vous ne l'avez pas déjà fait.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).
2. Exécutez la commande suivante pour créer une activation.

   Note

   région représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de *region* prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

   Linux

   ```
   aws ssm create-activation \
     --default-instance-name name \
     --iam-role iam-service-role-name \
     --registration-limit number-of-managed-instances \
     --region region \
     --tags "Key=key-name-1,Value=key-value-1" "Key=key-name-2,Value=key-value-2"
   ```

   Windows

   ```
   aws ssm create-activation ^
     --default-instance-name name ^
     --iam-role iam-service-role-name ^
     --registration-limit number-of-managed-instances ^
     --region region ^
     --tags "Key=key-name-1,Value=key-value-1" "Key=key-name-2,Value=key-value-2"
   ```

   PowerShell

   ```
   New-SSMActivation -DefaultInstanceName name `
     -IamRole iam-service-role-name `
     -RegistrationLimit number-of-managed-instances `
   ```

```
  -Region region `
  -Tag @{"Key"="key-name-1";"Value"="key-value-1"},@{"Key"="key-
name-2";"Value"="key-value-2"}
```

Voici un exemple de .

Linux

```
aws ssm create-activation \
  --default-instance-name MyWebServers \
  --iam-role service-role/AmazonEC2RunCommandRoleForManagedInstances \
  --registration-limit 10 \
  --region us-east-2 \
  --tags "Key=Environment,Value=Production" "Key=Department,Value=Finance"
```

Windows

```
aws ssm create-activation ^
      --default-instance-name MyWebServers ^
      --iam-role service-role/AmazonEC2RunCommandRoleForManagedInstances ^
      --registration-limit 10 ^
      --region us-east-2 ^
      --tags "Key=Environment,Value=Production" "Key=Department,Value=Finance"
```

PowerShell

```
New-SSMActivation -DefaultInstanceName MyWebServers `
  -IamRole service-role/AmazonEC2RunCommandRoleForManagedInstances `
  -RegistrationLimit 10 `
  -Region us-east-2 `
  -Tag
 @{"Key"="Environment";"Value"="Production"},@{"Key"="Department";"Value"="Finance"}
```

Si l'activation est créée avec succès, le système renvoie immédiatement un code d'activation et un ID.

Passez au Étape 6 : Installer Agent SSM pour un environnement hybride (Windows) (p. 62).

Étape 5 : Installer Agent SSM pour un environnement hybride (Linux)

Cette rubrique décrit comment installer Agent SSM sur des machines Linux dans un environnement hybride. Si vous prévoyez d'utiliser des machines Windows Server dans un environnement hybride, veuillez consulter l'étape précédente, Étape 6 : Installer Agent SSM pour un environnement hybride (Windows) (p. 62).

Important

Cette procédure concerne les serveurs et les machines virtuelles appartenant à un environnement sur site ou hybride. Pour télécharger et installer l'Agent SSM sur une instance EC2 pour Linux, reportez-vous à la section Installation et configuration de l'Agent SSM sur les instances EC2 pour Linux (p. 75).

Avant de commencer, recherchez le code d'activation et l'ID d'activation qui vous ont été envoyés à la fin de l'activation d'instance gérée, plus haut dans la rubrique Étape 4 : Créer une activation d'instance gérée pour un environnement hybride (p. 55). Vous indiquez le code et l'ID dans la procédure suivante.

Les URLs scripts suivants vous permettent Agent SSM de télécharger à partir de n'importe quelle région AWS. Si vous souhaitez télécharger l'agent à partir d'une région spécifique, copiez l'URL de votre système d'exploitation, puis remplacez *region* avec une valeur appropriée.

région représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de *region* prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

Par exemple, pour télécharger Agent SSM pour Amazon Linux , RHEL , et CentOS 64 bits à partir de l' SLES (us-west-1), utilisez l'URL suivante :Région USA Ouest (Californie du Nord)

```
https://s3.us-west-1.amazonaws.com/amazon-ssm-us-west-1/latest/linux_amd64/amazon-ssm-agent.rpm
```

Amazon Linux 2, Amazon Linux, RHEL, Oracle Linux, CentOS, and SLES

- Intel 64-bit (x86_64)

  https ://s3.*region*: amazonaws.com/amazon-ssm-*region*/dernière/linux_amd64/amazon-ssm-agent.rpm
- Intel 32-bit (x86)

  https ://s3.*region*: amazonaws.com/amazon-ssm-*region*/dernière/linux_386/amazon-ssm-agent.rpm
- ARM 64-bit (arm64)

  https ://s3.*region*: amazonaws.com/amazon-ssm-*region*/dernière/linux_arm64/amazon-ssm-agent.rpm

Ubuntu Server

- Intel 64-bit (x86_64)

  https ://s3.*region*: amazonaws.com/amazon-ssm-*region*/dernière/debian_amd64/amazon-ssm-agent.deb
- Intel 32-bit (x86)

  https ://s3.*region*: amazonaws.com/amazon-ssm-*region*/dernière/debian_386/amazon-ssm-agent.deb

Debian Server

- Intel 64-bit (x86_64)

  https ://s3.*region*: amazonaws.com/amazon-ssm-*region*/dernière/debian_amd64/amazon-ssm-agent.deb

Raspbian

- https ://s3.*region*: amazonaws.com/amazon-ssm-*region*/dernière/debian_arm/amazon-ssm-agent.deb

Pour installer Agent SSM sur des serveurs et VMs dans votre environnement hybride

1. Connectez-vous à un serveur ou une machine virtuelle de votre environnement hybride.
2. Copiez et collez l'un des blocs de commande suivants dans SSH. Remplacez les valeurs d'espace réservé par le code d'activation et l'ID d'activation générés lorsque vous créez une activation d'instance gérée, et par l'identifiant de la région AWS à Agent SSM partir de laquelle vous souhaitez effectuer le téléchargement, puis appuyez sur Enter.

   Notez que `sudo` n'est pas nécessaire si vous êtes un utilisateur racine.

   `région` représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de *`region`* prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

Amazon Linux, RHEL 6.x et CentOS 6.x

```
mkdir /tmp/ssm
curl https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/linux_amd64/amazon-ssm-
agent.rpm -o /tmp/ssm/amazon-ssm-agent.rpm
sudo yum install -y /tmp/ssm/amazon-ssm-agent.rpm
sudo stop amazon-ssm-agent
sudo amazon-ssm-agent -register -code "activation-code" -id "activation-id" -region
 "region"
sudo start amazon-ssm-agent
```

Amazon Linux 2, RHEL 7.x, et Oracle Linux 7.xCentOS

```
mkdir /tmp/ssm
curl https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/linux_amd64/amazon-ssm-
agent.rpm -o /tmp/ssm/amazon-ssm-agent.rpm
sudo yum install -y /tmp/ssm/amazon-ssm-agent.rpm
sudo systemctl stop amazon-ssm-agent
sudo amazon-ssm-agent -register -code "activation-code" -id "activation-id" -region
 "region"
sudo systemctl start amazon-ssm-agent
```

RHEL8.x et CentOS 8.x

```
mkdir /tmp/ssm
curl https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/linux_amd64/amazon-ssm-
agent.rpm -o /tmp/ssm/amazon-ssm-agent.rpm
sudo dnf install -y /tmp/ssm/amazon-ssm-agent.rpm
sudo systemctl stop amazon-ssm-agent
sudo amazon-ssm-agent -register -code "activation-code" -id "activation-id" -region
 "region"
sudo systemctl start amazon-ssm-agent
```

Debian Server

```
mkdir /tmp/ssm
wget https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/debian_amd64/amazon-
ssm-agent.deb -O /tmp/ssm/amazon-ssm-agent.deb
sudo dpkg -i /tmp/ssm/amazon-ssm-agent.deb
sudo service amazon-ssm-agent stop
```

```
sudo amazon-ssm-agent -register -code "activation-code" -id "activation-id" -region
 "region"
sudo service amazon-ssm-agent start
```

Raspbian

```
mkdir /tmp/ssm
sudo curl https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/debian_arm/amazon-
ssm-agent.deb -o /tmp/ssm/amazon-ssm-agent.deb
sudo dpkg -i /tmp/ssm/amazon-ssm-agent.deb
sudo service amazon-ssm-agent stop
sudo amazon-ssm-agent -register -code "activation-code" -id "activation-id" -region
 "region"
sudo service amazon-ssm-agent start
```

SLES

```
mkdir /tmp/ssm
sudo wget https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/linux_amd64/
amazon-ssm-agent.rpm
sudo rpm --install amazon-ssm-agent.rpm
sudo systemctl stop amazon-ssm-agent
sudo amazon-ssm-agent -register -code "activation-code" -id "activation-id" -region
 "region"
sudo systemctl enable amazon-ssm-agent
sudo systemctl start amazon-ssm-agent
```

Ubuntu Server

- Utilisation de packages .deb

  ```
  mkdir /tmp/ssm
  curl https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/debian_amd64/amazon-
  ssm-agent.deb -o /tmp/ssm/amazon-ssm-agent.deb
  sudo dpkg -i /tmp/ssm/amazon-ssm-agent.deb
  sudo service amazon-ssm-agent stop
  sudo amazon-ssm-agent -register -code "activation-code" -id "activation-id" -region
   "region"
  sudo service amazon-ssm-agent start
  ```

- Utilisation de packages Snap

  Vous n'avez pas besoin de spécifier une URL pour le téléchargement, car la commande `snap` télécharge automatiquement l'agent à partir de la boutique d'applications Snap à l'adresse https://snapcraft.io.

  ```
  sudo snap install amazon-ssm-agent --classic
  sudo systemctl stop snap.amazon-ssm-agent.amazon-ssm-agent.service
  sudo /snap/amazon-ssm-agent/current/amazon-ssm-agent -register -code "activation-code" -
  id "activation-id" -region "region"
  sudo systemctl start snap.amazon-ssm-agent.amazon-ssm-agent.service
  ```

  > **Important**
  >
  > Le canal candidat dans le magasin Snap contient la dernière version de Agent SSM ; pas le canal stable. Si vous souhaitez suivre les informations de Agent SSM version sur le canal des candidats, exécutez la commande suivante sur vos instances Ubuntu Server 18.04 et 16.04 LTS 64 bits.

```
sudo snap switch --channel=candidate amazon-ssm-agent
```

> **Note**
>
> Si vous voyez l'erreur suivante dans les journaux des erreurs de l'Agent SSM, cela signifie que l'ID de la machine n'a pas été conservé après un redémarrage :
> `Unable to load instance associations, unable to retrieve associations unable to retrieve associations error occurred in RequestManagedInstanceRoleToken: MachineFingerprintDoesNotMatch: Fingerprint does not match`
> Exécutez la commande suivante pour conserver l'ID de la machine après un redémarrage.

```
umount /etc/machine-id
systemd-machine-id-setup
```

La commande télécharge et installe Agent SSM sur le serveur ou la machine virtuelle de votre environnement hybride. La commande arrête l'Agent SSM, puis enregistre le serveur ou la machine virtuelle auprès du service SSM. Le serveur ou la machine virtuelle est désormais une instance gérée. Les instances EC2 configurées pour Systems Manager sont également des instances gérées. Pourtant, dans la console Systems Manager, vos instances sur site se distinguent des instances EC2 par le préfixe « mi- ».

> **Note**
>
> Vous pouvez annuler l'enregistrement d'une instance gérée en appelant l'action DeregisterManagedInstance d'API depuis l' ou AWS CLI l' .Outils pour Windows PowerShell Voici un exemple de commande de l'interface de ligne de commande :

```
aws ssm deregister-managed-instance --instance-id "mi-1234567890"
```

Passez au Étape 6 : Installer Agent SSM pour un environnement hybride (Windows) (p. 62).

Étape 6 : Installer Agent SSM pour un environnement hybride (Windows)

Cette rubrique décrit comment installer Agent SSM sur des machines Windows Server dans un environnement hybride. Si vous prévoyez d'utiliser des machines Linux dans un environnement hybride, veuillez consulter l'étape suivante, Étape 5 : Installer Agent SSM pour un environnement hybride (Linux) (p. 58).

> **Important**
>
> Cette procédure concerne les serveurs et les machines virtuelles appartenant à un environnement sur site ou hybride. Pour télécharger et installer l'Agent SSM sur une instance EC2 pour Windows Server, veuillez consulter Installation et configuration Agent SSM de sur les instances EC2 pourWindows Server (p. 100).

Avant de commencer, recherchez le code d'activation et l'ID d'activation qui vous ont été envoyés à la fin de l'activation d'instance gérée, plus haut dans la rubrique Étape 4 : Créer une activation d'instance gérée pour un environnement hybride (p. 55). Vous indiquez le code et l'ID dans la procédure suivante.

Pour installer Agent SSM sur des serveurs et VMs dans votre environnement hybride

1. Connectez-vous à un serveur ou une machine virtuelle de votre environnement hybride.
2. Ouvrez Windows PowerShell en mode élevé (administratif).

3. Copiez et collez le bloc de commande suivant dans WindowsPowerShell. Remplacez les valeurs d'espace réservé par le code d'activation et l'ID d'activation générés lorsque vous créez une activation d'instance gérée, et par l'identifiant de la région AWS à Agent SSM partir de laquelle vous souhaitez effectuer le téléchargement.

   région représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de *region* prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

   64-bit

```
$code = "activation-code"
$id = "activation-id"
$region = "region"
$dir = $env:TEMP + "\ssm"
New-Item -ItemType directory -Path $dir -Force
cd $dir
(New-Object System.Net.WebClient).DownloadFile("https://amazon-ssm-$region.s3.
$region.amazonaws.com/latest/windows_amd64/AmazonSSMAgentSetup.exe", $dir +
 "\AmazonSSMAgentSetup.exe")
Start-Process .\AmazonSSMAgentSetup.exe -ArgumentList @("/q", "/log",
 "install.log", "CODE=$code", "ID=$id", "REGION=$region") -Wait
Get-Content ($env:ProgramData + "\Amazon\SSM\InstanceData\registration")
Get-Service -Name "AmazonSSMAgent"
```

   32-bit

```
$code = "activation-code"
$id = "activation-id"
$region = "region"
$dir = $env:TEMP + "\ssm"
New-Item -ItemType directory -Path $dir -Force
cd $dir
(New-Object System.Net.WebClient).DownloadFile("https://amazon-ssm-$region.s3.
$region.amazonaws.com/latest/windows_386/AmazonSSMAgentSetup.exe", $dir +
 "\AmazonSSMAgentSetup.exe")
Start-Process .\AmazonSSMAgentSetup.exe -ArgumentList @("/q", "/log",
 "install.log", "CODE=$code", "ID=$id", "REGION=$region") -Wait
Get-Content ($env:ProgramData + "\Amazon\SSM\InstanceData\registration")
Get-Service -Name "AmazonSSMAgent"
```

4. Appuyez sur Enter.

La commande exécute les opérations suivantes :

- Télécharge et installe Agent SSM sur le serveur ou la machine virtuelle.
- Enregistre le serveur ou la machine virtuelle avec le service SSM.
- Renvoie à la demande une réponse semblable à ce qui suit :

```
    Directory: C:\Users\ADMINI~1\AppData\Local\Temp\2


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
d-----       07/07/2018   8:07 PM                ssm
{"ManagedInstanceID":"mi-008d36be46EXAMPLE","Region":"us-east-2"}

Status      : Running
```

```
Name        : AmazonSSMAgent
DisplayName : Amazon SSM Agent
```

Le serveur ou la machine virtuelle est désormais une instance gérée. Ces instances sont à présent identifiées avec le préfixe « mi- ». Vous pouvez visualiser les instances gérées sur la page Managed Instances (Instances gérées) dans la console Systems Manager, à l'aide de la commande AWS CLI describe-instance-information, ou de la commande d'API DescribeInstancePatches.

> **Note**
>
> Vous pouvez annuler l'enregistrement d'une instance gérée en appelant l'action DeregisterManagedInstance d'API depuis l' ou AWS CLI l' .Outils pour Windows PowerShell Voici un exemple de commande de l'interface de ligne de commande :
>
> ```
> aws ssm deregister-managed-instance --instance-id "mi-1234567890"
> ```

Démarrez avec AWS Systems Manager

Cette section vous permet de mieux connaître et d'utiliser AWS Systems Manager une fois que votre organisation a terminé les étapes de configuration de Configuration d'AWS Systems Manager (p. 28) ou Configuration d'AWS Systems Manager pour les environnements hybrides (p. 49).

Avant de commencer

Vous trouverez ci-dessous des renseignements utiles lors de la mise en route :

- la rubrique Présentation du AWS Systems Manager (p. 1) ;
- Amazon EC2 Getting Started Guide (si vous gérez des instances Amazon Elastic Compute Cloud (EC2) dans votre compte).
- comprendre les exigences de configuration de Systems Manager vous aide à résoudre les problèmes rencontrés lors de l'utilisation de Systems Manager, notamment concernant les autorisations ou la disponibilité des ressources :
  - Configuration d'AWS Systems Manager (p. 28)
  - Configuration d'AWS Systems Manager pour les environnements hybrides (p. 49)

Lorsque vous êtes prêt, passez aux étapes suivantes.

Rubriques

- Étape 1 : Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65)
- Étape 2 : Exercice pratique d'installation ou de mise à jour Agent SSM sur une instance (p. 67)
- Étape 3 : Essai des Systems Manager didacticiels et des procédures (p. 68)

Étape 1 : Installer ou mettre à niveau les outils de ligne de commande AWS

Cette rubrique est destinée aux utilisateurs qui bénéficient d'un accès par programme pour utiliser Systems Manager (ou tout autre service AWS), et qui souhaitent exécuter des commandes AWS CLI ou Outils AWS pour Windows PowerShell à partir de leur ordinateur local.

> Note
>
> L'accès par programme et l'accès par console sont des autorisations différentes qu'un administrateur de compte AWS peut accorder à un compte utilisateur. Un utilisateur peut bénéficier d'un type d'accès ou des deux. Pour plus d'informations, consultez la section Créer des utilisateurs et des groupes IAM non administrateurs pour Systems Manager (p. 30).

Pour de plus amples informations sur AWS CLI, veuillez consulter AWS Command Line Interface Guide de l'utilisateur. Pour de plus amples informations sur Outils AWS pour Windows PowerShell, veuillez consulter Outils AWS pour Windows PowerShell Guide de l'utilisateur.

Pour de plus amples informations sur toutes les commandes Systems Manager que vous pouvez exécuter à l'aide des outils AWS CLI, veuillez consulter la section Systems Manager du AWS CLI Command

Reference. Pour de plus amples informations sur toutes les commandes Systems Manager que vous pouvez exécuter à l'aide des outils Outils AWS pour PowerShell, veuillez consulter la section Systems Manager du Outils AWS pour PowerShell Cmdlet Reference.

> **Important**
>
> À partir du 10 janvier 2020, l'AWS CLI versions 1.17 et ultérieures ne prendra plus en charge Python 2.6 ou Python 3.3. Après cette date, le programme d'installation de l'AWS CLI nécessitera Python 2.7, Python 3.4 ou une version ultérieure pour installer l'AWS CLI. Pour de plus amples informations, veuillez consulter Utilisation de l'AWS CLI version 1 avec Python 2.6 ou Python 3.3 dans le IAM Guide de l'utilisateur.

AWS CLI

Installation ou mise à niveau, puis configuration de l'AWS CLI

1. Suivez les instructions fournies dans la section Installation de l'AWS Command Line Interface dans le AWS Command Line Interface Guide de l'utilisateur pour installer ou mettre à niveau l'AWS CLI sur votre ordinateur local.

   > **Tip**
   >
   > De nouvelles fonctionnalités sont fréquemment ajoutées à l'AWS CLI. Mettez à niveau (réinstallez) régulièrement l'interface de ligne de commande afin de vous assurer que vous avez accès à toutes les nouvelles fonctionnalités.

2. Pour configurer l'AWS CLI, consultez Configuration de l'AWS Command Line Interface dans le AWS Command Line Interface Guide de l'utilisateur.

   Au cours de cette étape, vous devez spécifier des informations d'identification qu'un administrateur AWS de votre organisation vous a données, au format suivant :

   ```
   AWS Access Key ID: AKIAIOSFODNN7EXAMPLE
   AWS Secret Access Key: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
   ```

   > **Important**
   >
   > Lorsque vous configurez l'interface AWS CLI, vous êtes invité à spécifier une région AWS. Choisissez l'une des régions prises en charge répertoriées pour Systems Manager dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services. Si nécessaire, vérifiez au préalable quelle région vous devez choisir avec un administrateur de votre compte AWS.

   Pour de plus amples informations sur les clés d'accès, consultez Gestion des clés d'accès pour les utilisateurs IAM dans le IAM Guide de l'utilisateur.

3. Pour vérifier l'installation ou la mise à niveau, exécutez la commande suivante à partir de l'AWS CLI :

   ```
   aws ssm help
   ```

   En cas de succès, cette commande affiche la liste des commandes Systems Manager disponibles.

Outils AWS pour PowerShell

Installation ou mise à niveau, puis configuration de l'Outils AWS pour Windows PowerShell

1. Suivez les instructions de la section Configuration d'Outils AWS pour Windows PowerShell ou Outils AWS pour PowerShell Core dans le Outils AWS pour Windows PowerShell Guide de

l'utilisateur pour installer ou mettre à niveau Outils AWS pour PowerShell sur votre ordinateur local.

> **Tip**
>
> De nouvelles fonctionnalités sont fréquemment ajoutées à Outils AWS pour PowerShell. Mettez à niveau (réinstallez) régulièrement les outils Outils AWS pour PowerShell afin de vous assurer d'avoir accès à toutes les nouvelles fonctionnalités.

2. Pour configurer Outils AWS pour PowerShell, veuillez consulter Utilisation des informations d'identification AWS dans le Outils AWS pour Windows PowerShell Guide de l'utilisateur.

   Au cours de cette étape, vous devez spécifier à l'aide de la commande suivante laes informations d'identification qu'un administrateur AWS de votre organisation vous a données.

```
Set-AWSCredential `
-AccessKey AKIAIOSFODNN7EXAMPLE `
-SecretKey wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY `
-StoreAs MyProfileName
```

> **Important**
>
> Lorsque vous configurez Outils AWS pour PowerShell, vous pouvez exécuter `Set-DefaultAWSRegion` pour spécifier une région AWS. Choisissez l'une des régions prises en charge répertoriées pour Systems Manager dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services. Si nécessaire, vérifiez au préalable quelle région vous devez choisir avec un administrateur de votre compte AWS.

   Pour de plus amples informations sur les clés d'accès, veuillez consulter Gestion des clés d'accès pour les utilisateurs IAM dans le IAM Guide de l'utilisateur.

3. Pour vérifier l'installation ou la mise à niveau, exécutez la commande suivante à partir d'Outils AWS pour PowerShell :

```
Get-AWSCmdletName -Service SSM
```

   En cas de succès, cette commande affiche la liste des cmdlets Systems Manager disponibles.

Étape 2 : Exercice pratique d'installation ou de mise à jour Agent SSM sur une instance

AWS Systems Manager Agent (Agent SSM) est un logiciel Amazon qui peut être installé et configuré sur une instance EC2, un serveur sur site ou une machine virtuelle (VM). L'Agent SSM permet à Systems Manager de mettre à jour, de gérer et de configurer ces ressources. L'agent traite les demandes du service Systems Manager dans le cloud AWS et les exécute ensuite comme spécifié dans la demande. Puis, Agent SSM renvoie les informations relatives à l'état et à l'exécution au service Systems Manager à l'aide de Amazon Message Delivery Service (préfixe du service : `ec2messages`).

Dans cette tâche, vous devez installer ou mettre à jour l'Agent SSM sur une instance EC2.

> **Note**
>
> Si vous utilisez vos propres serveurs sur site ou VMs , consultez les rubriques suivantes :
>
> - Installer Agent SSM pour un environnement hybride (Windows) (p. 62)
> - Installer Agent SSM pour un environnement hybride (Linux) (p. 58)

Prerequisites

Un profil d'instance pour Systems Manager doit déjà être attaché à l'instance EC2 que vous mettez à jour. Consultez les rubriques suivantes en fonction des besoins afin de respecter cette exigence :

- Créer un profil d'instance EC2 pour Systems Manager : Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34)
- Attacher le profil d'instance à une instance EC2 lorsque vous créez l'instance : Attachement d'un profil d'instance IAM à une instance EC2 (p. 40)
- Attachez le profil d'instance à une instance EC2 existante : Attachement d'un IAM rôle à une instance dans le Guide Amazon EC2 de l'utilisateur

Instance Windows Server

Pour vous entraîner à installer ou à mettre à jour l'Agent SSM sur une instance EC2 pour Windows Server, suivez les étapes décrites à la section Installation manuelle Agent SSM sur les instances EC2 pourWindows Server (p. 100).

Instance Linux

Pour vous entraîner à installer ou à mettre à jour l'Agent SSM sur une instance EC2 pour Linux, suivez les étapes adaptées au type de votre système d'exploitation Linux dans Installer manuellement l'Agent SSM sur les instances EC2 pour Linux (p. 76).

Étape 3 : Essai des Systems Manager didacticiels et des procédures

Cette rubrique vous permet d'accéder aux didacticiels, procédures et tâches de base qui vous permettront d'apprendre à utiliser Systems Manager.

Systems Manager étant un ensemble de plusieurs fonctionnalités, aucun didacticiel ou procédure ne peut présenter l'ensemble du service. Nous vous fournissons donc des liens vers des ressources selon la fonctionnalité étudiée.

Dans la plupart des cas, il n'est pas nécessaire d'effectuer des tâches d'installation ou de configuration supplémentaires avant de commencer. Vous pouvez effectuer toutes les tâches si vous disposez des autorisations nécessaires et, si besoin, de l'accès à une ou plusieurs instances gérées.

Dans certains cas, une configuration, une installation ou une expérience supplémentaire avec Systems Manager est requise si vous souhaitez suivre un didacticiel ou une procédure. Nous avons attribué à ces didacticiels et procédures le statut « Avancé ».

Compliance

La fonctionnalité Conformité de la configuration AWS Systems Manager (p. 778) analyse votre parc d'instances gérées afin de rechercher des incohérences de conformité et de configuration de correctifs.

- Procédure pas à pas de Conformité de la configuration (AWS CLI) (p. 787)

Fonctionnalité Exécuter la commande

La AWS Systems Manager Fonctionnalité Exécuter la commande (p. 941) fonctionnalité vous offre une gestion à distance, sûre et sécurisée, de vos instances à grande échelle sans connexion à vos serveurs, sans avoir besoin d'hôtes bastion, de SSH ou d' PowerShell à distance. Il fournit un moyen

simple d'automatiser les tâches administratives courantes dans les groupes d'instances, telles que les modifications de registre, la gestion des utilisateurs et les installations de logiciels et de correctifs.

- Procédure : Utilisation de l' AWS CLI avecFonctionnalité Exécuter la commande (p. 971)
- Procédure : Utilisation de l' Outils AWS pour Windows PowerShell avecFonctionnalité Exécuter la commande (p. 977)

Session Manager

La fonctionnalité AWS Systems Manager Session Manager (p. 869) vous permet de gérer vos instances EC2 via un shell interactif basé sur navigateur et en un clic, ou via l'AWS CLI sans qu'il soit nécessaire d'ouvrir les ports entrants, de maintenir les hôtes bastions ou de gérer les clés SSH.

- Utilisation des Session Manager (p. 917)

Distributor

La AWS Systems Manager Distributor (p. 1174) fonctionnalité vous permet d'empaqueter vos propres logiciels ou de trouver les packages logiciels d'agent fournis par AWS, tels que AmazonCloudWatchAgent - à installer sur les instances Systems Manager gérées.

- Créer un package (p. 1182)
- Étape 4 : Ajouter un package à Distributor (p. 1191)

Gestionnaire de correctifs

La AWS Systems Manager Gestionnaire de correctifs (p. 1047) fonctionnalité vous aide à sélectionner et à déployer automatiquement des correctifs logiciels et de système d'exploitation sur de grands groupes d'instances EC2 ou sur des serveurs sur site et VMs .

- Utilisation de références de correctifs personnalisées (p. 1127)
- Création d'un groupe de correctifs (p. 1137)
- Procédure : Appliquer un correctif à un environnement serveur (AWS CLI) (p. 1168)

Fenêtres de maintenance

La fonctionnalité AWS Systems Manager Fenêtres de maintenance (p. 688) vous permet de définir un planning d'exécution d'actions potentiellement perturbatrices sur vos instances gérées, telles que l'application de correctifs à un système d'exploitation, la mise à jour de pilotes ou l'installation de logiciels ou de correctifs.

- Didacticiel : Créer et configurer une fenêtre de maintenance (AWS CLI) (p. 717)
- Didacticiel : Mettre à jour une fenêtre de maintenance (AWS CLI) (p. 754)
- Didacticiel : Affichage des informations sur les fenêtres de maintenance (AWS CLI) (p. 740)
- Didacticiel : Afficher les informations sur les tâches et les exécutions de tâches (AWS CLI) (p. 751)

State Manager

La AWS Systems Manager State Manager (p. 987) capacité vous permet de maintenir une configuration cohérente de vos instances EC2 ou des serveurs sur site et VMs , dans un état que vous définissez. Avec State Manager, vous pouvez contrôler les détails de configuration tels que les configurations des serveurs, les définitions anti-virus, les paramètres des pare-feu, etc.

- Création d'associations qui exécutent des fichiers MOF (p. 1019)

- Mise à jour automatique d'Agent SSM (interface de ligne de commande) (p. 1043)
- Procédure : Mise à jour automatique des pilotes PV sur les instances EC2 pour Windows Server (console) (p. 1045)

Documents

La fonctionnalité Documents AWS Systems Manager (p. 1211) vous permet de créer et de gérer des documents SSM. Un document SSM définit les actions que Systems Manager effectue sur vos instances gérées. Systems Manager inclut plus d'une dizaine de documents préconfigurés que vous pouvez utiliser en spécifiant des paramètres lors de l'exécution. Les documents utilisent JavaScript le format JSON (Object Notation) ou YAML et incluent les étapes et les paramètres que vous spécifiez.

- Créer un document SSM (console) (p. 1269)
- Créer un document SSM (ligne de commande) (p. 1270)

Parameter Store

La fonctionnalité AWS Systems Manager Parameter Store (p. 218) fournit également un magasin centralisé permettant de gérer les données de configuration, qu'il s'agisse de données en texte brut (comme des chaînes de base de données) ou de codes secrets (comme des mots de passe). Vous pouvez ainsi séparer les codes secrets et les données de configuration du code. Les paramètres peuvent être balisés et hiérarchisés, ce qui vous aide à les gérer plus facilement.

- Créer un paramètre (console) (p. 245)
- Créer un paramètre Systems Manager (AWS CLI) (p. 246)
- Utilisation des hiérarchies de paramètres (p. 285)
- Avancé : Créer un SecureString paramètre et joindre une instance à un domaine (PowerShell) (p. 300)

Inventory

La fonctionnalité Inventaire AWS Systems Manager (p. 791) recueille des informations sur vos instances et le logiciel dont elles sont équipées pour vous aider à comprendre les configurations de votre système et les applications qui y sont installées.

- Avancé : Procédure : Attribuer des métadonnées d'inventaire personnalisé à une instance (p. 838)
- Avancé : Procédure : Configurer vos instances gérées pour l'inventaire à l'aide de l'interface de ligne de commande (p. 839)
- Avancé : Procédure : Utiliser la synchronisation de données de ressources pour regrouper les données d'inventaire (p. 843)

Automation

La fonctionnalité Automatisation AWS Systems Manager (p. 304) vous permet d'automatiser en toute sécurité des opérations et des tâches de gestion entre les ressources AWS. Vous pouvez automatiser des tâches informatiques courantes, effectuer des tâches perturbatrices par lots en toute sécurité, simplifier des tâches complexes, améliorer les opérations de sécurité, et utiliser les scripts de configuration stockés pour partager les bonnes pratiques avec le reste de votre organisation.

- Avancé : Procédure : Appliquer un correctif à une AMI Linux (console) (p. 633)
- Avancé : Procédure : Appliquer un correctif à une AMI Linux (AWS CLI) (p. 636)
- Avancé : Procédure : Appliquer un correctif à une AMI Windows Server (p. 640)
- Avancé : Procédure : Simplification de l'application de correctifs à l'AMI avec Automation, AWS Lambda et Parameter Store (p. 645)

- Avancé : Procédure : Appliquer des correctifs à une AMI et mettre à jour un groupe Auto Scaling (p. 651)
- Avancé : Procédure : Exécuter l'EC2Rescueoutil sur des instances inaccessibles (p. 656)
- Avancé : Procédure : Réinitialiser les mots de passe et les clés SSH sur les instances EC2 (p. 661)
- Avancé : Procédure : Utilisation d'Automation avec Jenkins (p. 667)

Utilisation des Agent SSM

AWS Systems Manager Agent (Agent SSM) est un logiciel Amazon qui peut être installé et configuré sur une instance EC2, un serveur sur site ou une machine virtuelle (VM). L'Agent SSM permet à Systems Manager de mettre à jour, de gérer et de configurer ces ressources. L'agent traite les demandes du service Systems Manager dans le cloud AWS et les exécute ensuite comme spécifié dans la demande. Puis, Agent SSM renvoie les informations relatives à l'état et à l'exécution au service Systems Manager à l'aide de Amazon Message Delivery Service (préfixe du service : `ec2messages`).

Si vous surveillez le trafic, vous verrez vos instances Amazon Elastic Compute Cloud (Amazon EC2) et tous les serveurs sur site ou VMs dans votre environnement hybride, en communiquant avec des `ec2messages.*` points de terminaison. Pour plus d'informations, consultez la section Référence : ec2messages, ssmmessages et autres appels d'API (p. 1399). Pour de plus amples informations sur le portage de journaux de Agent SSM vers Amazon CloudWatch Logs, consultez Surveillance des AWS Systems Manager (p. 1326).

Gardez l'Agent SSM à jour

Une version mise à jour de l'Agent SSM est publiée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Si une version antérieure de l'agent est en cours d'exécution sur une instance, certains processus de l'Agent SSM peuvent échouer. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir l'Agent SSM à jour sur vos instances. Pour de plus amples informations, veuillez consulter Automatisation des mises à jour versAgent SSM (p. 109). Pour être alerté des mises à jour de Agent SSM, abonnez-vous à la page Agent SSM Release Notes sur GitHub.

> Note
>
> Amazon Machine Images(AMIs) qui inclut Agent SSM par défaut peut prendre jusqu'à deux semaines pour être mis à jour avec la version la plus récente d' Agent SSM . Nous vous recommandons de configurer encore plus fréquemment des mises à jour automatiques de l'Agent SSM.
> Agent SSMLes versions mises à jour d' sont déployées dans de nouvelles AWS régions à différents moments. C’est pourquoi, vous pouvez recevoir l'erreur « Non pris en charge sur la plate-forme actuelle » ou « Mise à jour d'amazon-ssm-agent vers une ancienne version, veuillez activer autoriser le retour à une version plus ancienne pour continuer » lorsque vous essayez de déployer une nouvelle version de l’Agent SSM dans une région.

Agent SSMet le Instance Metadata Service (IMDS)

Systems Managers'appuie sur les métadonnées d'instance EC2 pour fonctionner correctement. Systems Manager peut accéder aux métadonnées d'instance à l'aide de la version 1 ou de la version 2 du kit Instance Metadata Service (IMDSv1 et IMDSv2. Pour plus d'informations, consultez Métadonnées d'instance et données utilisateur dans le Amazon EC2 Guide de l'utilisateur pour les instances Linux .

Priorité des informations d'identification de l’Agent SSM

Quand Agent SSM est installé sur une instance, il nécessite des autorisations pour communiquer avec le Systems Manager service. Sur les instances EC2, ces autorisations sont fournies dans un profil d'instance attaché à l'instance. Sur une instance hybride, obtient Agent SSM normalement les autorisations nécessaires à partir du fichier d'informations d'identification partagé, situé dans `/root/.aws/credentials` (Linux) ou `%USERPROFILE%\.aws\credentials` dans (Windows. Les autorisations nécessaires sont ajoutées à ce fichier au cours du processus d'activation hybride.

Toutefois, dans de rares cas, une instance peut se retrouver avec des autorisations ajoutées à plusieurs emplacements où Agent SSM vérifie les autorisations pour exécuter ses tâches.

A titre d'exemple, vous pouvez configurer une instance pour qu'elle soit gérée par Systems Manager . Pour une instance EC2, cette configuration inclut l'attachement d'un profil d'instance. Pour un serveur sur site ou une machine virtuelle (VM), cela signifie que les informations d'identification sont fournies via le processus d'activation hybride. Mais vous décidez ensuite d'utiliser également cette instance pour les tâches de développeur ou d'utilisateur final et d'installer l' AWS CLI sur celle-ci. Cette installation entraîne l'ajout d'autorisations supplémentaires à un fichier d'informations d'identification sur l'instance.

Lorsque vous exécutez une Systems Manager commande sur l'instance, Agent SSM peut essayer d'utiliser des informations d'identification différentes de celles qu'il s'attend à utiliser, par exemple à partir d'un fichier d'informations d'identification au lieu d'un profil d'instance. Cela est dû au fait qu' Agent SSM recherche les informations d'identification dans l'ordre prescrit pour la chaîne du fournisseur d'informations d'identification par défaut.

> **Note**
>
> Sous Linux, Agent SSM s'exécute en tant qu'utilisateur racine. Par conséquent, les variables d'environnement et le fichier d'informations d'identification que Agent SSM recherche dans ce processus sont celles de l'utilisateur racine uniquement (`/root/.aws/credentials`). Agent SSM ne regarde pas les variables d'environnement ou le fichier d'informations d'identification des autres comptes d'utilisateur sur l'instance lors de la recherche des informations d'identification.

La chaîne de fournisseur par défaut recherche les informations d'identification dans l'ordre suivant :

1. Variables d'environnement, si configurées (`AWS_ACCESS_KEY_ID` et `AWS_SECRET_ACCESS_KEY`).
2. Fichier d'informations d'identification partagé (`$HOME/.aws/credentials` pour Linux ou `%USERPROFILE%\.aws\credentials` pour Windows) avec les autorisations fournies par, par exemple, une activation hybride ou une AWS CLI installation.
3. Un rôle AWS Identity and Access Management (IAM) pour les tâches si une application qui utilise une définition de tâche Amazon Elastic Container Service (Amazon ECS) ou une opération dRunTask'API est présente.
4. Profil d'instance attaché à une Amazon EC2 instance.

Pour obtenir des informations connexes, consultez les rubriques suivantes :

- Profils d'instance pour les instances EC2 – Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34) etAttachement d'un profil d'instance IAM à une instance EC2 (p. 40)
- Activations hybrides–Créer une activation d'instance gérée pour un environnement hybride (p. 55)
- AWS CLIcredentials – Paramètres de configuration et du fichier d'informations d'identification dans l'AWS Command Line Interface Guide de l'utilisateur
- Chaîne de fournisseur d'informations d'identification par défaut – Spécification des informations d'identification dans l'Manuel du développeur Kit SDK AWS pour Go

  > **Note**
  >
  > Cette rubrique du Manuel du développeur Kit SDK AWS pour Go décrit la chaîne de fournisseur par défaut en termes de . Toutefois, les mêmes principes s'appliquent à l'évaluation des informations d'identification pour Kit SDK pour Go .Agent SSM

À propos du compte local ssm-user

A partir de la version 2.3.50.0 d' Agent SSM , l'agent crée un compte utilisateur local appelé `ssm-user` et l'ajoute à `/etc/sudoers` (Linux) ou au groupe Administrateurs (Windows. Sur les versions de l'agent antérieures à la version 2.3.612.0, le compte est créé la première fois qu'Agent SSMil démarre ou redémarre après l'installation. Sur la version 2.3.612.0 et version ultérieure, le compte `ssm-user` est créé la première fois qu'une session est démarrée sur une instance. Ce `ssm-user` est l'utilisateur du système d'exploitation par défaut lorsqu'une session Session Manager est démarrée. Vous pouvez modifier les

autorisations de l'utilisateur `ssm-user` en le déplaçant vers un groupe ayant moins de privilèges ou en modifiant le fichier `sudoers`. Le compte `ssm-user` n'est pas supprimé du système lorsque l'Agent SSM est désinstallé.

Sur Windows Server, l'Agent SSM gère la définition d'un nouveau mot de passe pour le compte `ssm-user` lorsque chaque session commence. Aucun mot de passe n'est défini pour `ssm-user` sur des instances gérées Linux.

À partir de la version Agent SSM 2.3612.0, le compte `ssm-user` n'est pas créé automatiquement sur les ordinateurs Windows Server qui sont utilisés en tant que contrôleurs de domaine. Pour utiliser Session Manager sur un contrôleur de domaine Windows Server, vous devez créer le compte `ssm-user` manuellement s'il n'est pas déjà présent.

> **Important**
>
> Pour que le compte ssm-user soit créée, le profil d'instance attaché à l'instance doit fournir les autorisations requises. Pour plus d'informations, consultez la section Vérifier ou créer un profil d'instance IAM avec des autorisations Session Manager (p. 876).

AMIsavec Agent SSM préinstallé

Agent SSMest préinstallé par défaut sur les éléments suivants Amazon Machine Images (AMIs) :

- Amazon Linux
- Amazon Linux 2
- AMIs optimisées pour ECS Amazon Linux 2
- Ubuntu Server 16.04, 18.04 et 20.04
- AMIs Windows Server 2008-2012 R2 publiées à partir de novembre 2016
- Windows Server 2016 et 2019

> **Note**
>
> L'Agent SSM n'est pas installé sur toutes les AMIs basées sur Amazon Linux ou Amazon Linux 2. Par exemple, l'Agent SSM n'est pas préinstallé sur les AMIs optimisées pour EKS basées sur Amazon Linux 2.

Vous devez installer manuellement Agent SSM sur les instances EC2 créées à partir d'autres LinuxAMIs. Vous devez également installer manuellement Agent SSM sur les serveurs sur site ou VMs dans votre environnement hybride. Pour plus d'informations, consultez la section Configuration d'AWS Systems Manager pour les environnements hybrides (p. 49).

Agent SSMsurGitHub

Le code source pour Agent SSM est disponible sur GitHub afin que vous puissiez adapter l'agent pour répondre à vos besoins. Nous vous conseillons d'envoyer des requêtes d'extraction pour les modifications que vous souhaitez inclure. Cependant, Amazon Web Services ne fournit actuellement aucun support pour l'exécution de copies modifiées de ce logiciel.

Table des matières

- Agent SSMversion 3 (p. 75)
- Installation et configuration de l'Agent SSM sur les instances EC2 pour Linux (p. 75)
- Installation et configuration Agent SSM de sur les instances EC2 pourWindows Server (p. 100)
- Vérification de Agent SSM l'état et démarrage de l'agent (p. 104)
- Vérification du numéro de Agent SSM version (p. 106)
- Affichage des Agent SSM journaux (p. 107)

- Restriction de l'accès aux commandes de niveau racine viaAgent SSM (p. 108)
- Automatisation des mises à jour versAgent SSM (p. 109)
- Abonnement aux notifications Agent SSM (p. 111)
- À propos des autorisations minimales relatives aux compartiments S3 pour l'Agent SSM (p. 112)
- Dépannage de Agent SSM (p. 115)

Agent SSMversion 3

Le 21 septembre 2020, AWS Systems Manager version 3.0 Agent SSM publiée. Notez les informations importantes suivantes sur cette version :

- La version 3.0 est rétrocompatible avec la version 2.x.
- Si vous avez configuré vos instances gérées pour une mise à jour automatique à l'aide de la version cible Agent SSM (configuration par défaut pour la mise à jour automatique), met `$LATEST` automatiquement à jour l'agent sur vos instances vers la version 3.0 et supprime la version 2.x.Systems Manager

  Si vous téléchargez manuellement Agent SSM , le système installe la version 2.x.
- Lorsque vous effectuez une mise à jour vers la version 3.0, le système renomme votre instance gérée `amazon-ssm-agent` en `.ssm-agent-worker` La mise à jour installe ensuite un nouveau fichier binaire nommé `amazon-ssm-agent` . Les nouvelles fonctions binaires en tant que gestionnaire de processus pour `ssm-agent-worker` . Le `ssm-agent-worker` fichier binaire communique directement avec Systems Manager pour traiter les demandes.

  > **Important**
  >
  > Les clients qui s'exécutent Agent SSM dans des environnements sécurisés doivent ajouter `ssm-agent-worker` à la liste des autorisations dans votre environnement informatique avant la mise à niveau.
- La version 2.x est toujours prise en charge, mais elle sera obsolète.
- Amazon Machine Images(AMIs) installe la version 2.x jusqu'à ce que la version 2.x soit obsolète.
- Avec la version 3.0, les événements Agent SSM de démarrage et de mise à jour sont enregistrés sur l'instance. Pour plus d'informations sur l'affichage des fichiers Agent SSM journaux, consultez Affichage des Agent SSM journaux (p. 107) .
- Vous pouvez utiliser Amazon CloudWatch pour effectuer des actions sur Agent SSM lorsque le système enregistre un événement de démarrage ou de mise à jour. Pour plus d'informations, consultez Créer une CloudWatch alarme basée sur un seuil statique dans le Guide de l'utilisateur Amazon CloudWatch .
- Avec la version 3.0, il n'y a pas de modification de ce qui suit :
  - Les autorisations minimales AWS Identity and Access Management (IAM), la chaîne d'informations d'identification ou le processus `ssm-user` de création
  - Plateformes prises en charge, emplacement des journaux ou journalisation de débogage
  - Prise en charge du traitement des commandes ou du plug-in SSM
  - Processus de configuration du proxy
  - Windows Clés de registre

Installation et configuration de l'Agent SSM sur les instances EC2 pour Linux

L'Agent SSM traite les demandes Systems Manager et configure la machine comme spécifié dans la demande. Utilisez les procédures suivantes pour installer, configurer ou désinstaller l'Agent SSM.

Important

Agent SSMest préinstallé, par défaut, sur les éléments suivants Amazon Machine Images (AMIs) :

- Amazon Linux
- Amazon Linux 2
- Amazon Linux 2 ECS-Optimized AMIs
- Ubuntu Server 16.04, 18.04, and 20.04

Note

Agent SSM is not installed on all AMIs based on Amazon Linux or Amazon Linux 2. For example, Agent SSM is not preinstalled on EKS-Optimized AMIs based on Amazon Linux 2.

Vous devez installer manuellement Agent SSM sur les instances EC2 créées à partir d'autres LinuxAMIs.

Le code source pour Agent SSM est disponible sur GitHub afin que vous puissiez adapter l'agent pour répondre à vos besoins. Nous vous conseillons d'envoyer des requêtes d'extraction pour les modifications que vous souhaitez inclure. Toutefois, AWS ne fournit actuellement aucun support pour l'exécution de copies modifiées de ce logiciel.

Note

Pour afficher des informations sur les différentes versions de Agent SSM, consultez les notes de mise à jour.

Rubriques

- Installer manuellement l'Agent SSM sur les instances EC2 pour Linux (p. 76)
- Configurer Agent SSM pour utiliser un proxy (p. 96)
- Désinstaller Agent SSM sur les instances Linux (p. 99)

Installer manuellement l'Agent SSM sur les instances EC2 pour Linux

Avant d'installer manuellement Agent SSM sur un système d'exploitation Amazon EC2 Linux, veuillez consulter les détails importants suivants.

- Agent SSMest installé par défaut sur les instances EC2 suivantes et Amazon Machine Images (AMIs) :
  - Amazon Linux
  - Amazon Linux 2
  - Amazon Linux 2 ECS-Optimized AMIs
  - Ubuntu Server 16.04, 18.04, and 20.04

    Note

    Agent SSM is not installed on all AMIs based on Amazon Linux or Amazon Linux 2. For example, Agent SSM is not preinstalled on EKS-Optimized AMIs based on Amazon Linux 2.
- Les procédures manuelles vous permettent Agent SSM d'installer à partir de n'importe quelle AWS région. Si vous souhaitez télécharger l'agent à partir d'une région spécifique, consultez « Télécharger Agent SSM à partir d'une AWS région spécifique » plus loin dans cette rubrique. Cette section décrit

comment modifier l'URL que vous spécifiez dans la procédure manuelle. Cette section ne décrit pas comment télécharger et installer l'agent. Vous devez choisir l'un des liens du système d'exploitation, puis remplacer l'URL pour télécharger à partir d'une région spécifique.

- Si vous avez besoin d'installer l'agent sur un serveur sur site ou sur une machine virtuelle (VM) afin qu'il puisse être utilisé avec Systems Manager, consultez Installer Agent SSM pour un environnement hybride (Linux) (p. 58).
- Une version mise à jour de l'Agent SSM est publiée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Si une version antérieure de l'agent est en cours d'exécution sur une instance, certains processus de l'Agent SSM peuvent échouer. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir l'Agent SSM à jour sur vos instances. Pour de plus amples informations, veuillez consulter Automatisation des mises à jour versAgent SSM (p. 109). Pour être alerté des mises à jour de Agent SSM, abonnez-vous à la page Agent SSM Release Notes sur GitHub.

Choisir votre système d'exploitation

Choisissez un lien pour afficher la procédure d'installation manuelle de Agent SSM sur le système d'exploitation spécifié.

- Amazon Linux (p. 77)
- Amazon Linux 2 (p. 79)
- CentOS (p. 80)
- Debian Server (p. 84)
- Oracle Linux (p. 86)
- Red Hat Enterprise Linux (p. 87)
- SUSE Linux Enterprise Server (p. 90)
- Ubuntu Server (p. 91)

Installation manuelle de Agent SSM sur les instances Amazon Linux

Connectez-vous à votre instance Amazon Linux et effectuez les opérations suivantes pour installer l'Agent SSM.

> **Note**
>
> Si vous utilisez une commande `yum` pour mettre à jour Agent SSM sur une instance gérée après l'installation ou la mise à niveau de l'agent à l'aide d'un document SSM`AWS-UpdateSSMAgent`, le message suivant pourrait s'afficher : « Avertissement : RPMDB a été modifiée hors de yum ». Ce message est attendu et peut être ignoré en toute sécurité.

Effectuez ces étapes sur chaque instance qui utilisera Systems Manager .

> **Note**
>
> Agent SSMest installé, par défaut, sur Amazon Linux une base AMIs de base datée de 2017.09 et ultérieures. Agent SSM est également installé, par défaut, sur les Amazon Linux 2 AMI. Vous devez installer manuellement Agent SSM sur les autres versions de Linux.
> Les instances créées depuis une AMI Amazon Linux qui utilisent un proxy doivent exécuter une version actuelle du module Python `requests` afin de prendre en charge les opérations du Gestionnaire de correctifs. Pour de plus amples informations, veuillez consulter Mettre à niveau le module de demandes Python sur Amazon Linux les instances qui utilisent un serveur proxy (p. 98).

Pour installer l'Agent SSM sur Amazon Linux

Utilisez l'une des commandes suivantes pour télécharger et exécuter le programme d'installation de l'Agent SSM.

région représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de *region* prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

Intel 64-bit (x86_64)

```
sudo yum install -y https://s3.region.amazonaws.com/amazon-ssm-region/latest/linux_amd64/
amazon-ssm-agent.rpm
```

Intel 32-bit (x86)

```
sudo yum install -y https://s3.region.amazonaws.com/amazon-ssm-region/latest/linux_386/
amazon-ssm-agent.rpm
```

ARM 64-bit (arm64)

```
sudo yum install -y https://s3.region.amazonaws.com/amazon-ssm-region/latest/linux_arm64/
amazon-ssm-agent.rpm
```

> **Note**
>
> Si vous ne parvenez pas à télécharger l'agent à partir de la région que vous spécifiez, utilisez l'une des URL globales ci-dessous. Notez que même si les URL suivantes indiquent « ec2-downloads-windows », ce sont les URL correctes pour les systèmes d'exploitation Linux.
>
> - Intel 64-bit (x86_64)
>
> ```
> https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/linux_amd64/
> amazon-ssm-agent.rpm
> ```
>
> - Intel 32-bit (x86)
>
> ```
> https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/linux_386/amazon-
> ssm-agent.rpm
> ```
>
> - ARM 64-bit (arm64)
>
> ```
> https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/linux_arm64/
> amazon-ssm-agent.rpm
> ```

Après avoir installé l'agent, exécutez la commande suivante pour déterminer si l' Agent SSM est en cours d'exécution. La commande doit renvoyer le message `amazon-ssm-agent is running`.

```
sudo status amazon-ssm-agent
```

Si la commande précédente renvoie le message `amazon-ssm-agent is stopped` , exécutez les commandes suivantes :

```
sudo start amazon-ssm-agent
sudo status amazon-ssm-agent
```

> **Important**
>
> Une version mise à jour de l'Agent SSM est publiée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Si une version antérieure de l'agent est en cours d'exécution sur une instance, certains processus de l'Agent SSM peuvent échouer. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir l'Agent SSM à jour sur vos instances. Pour de plus amples informations, veuillez consulter Automatisation des mises à jour versAgent SSM (p. 109). Pour être alerté des mises à jour de Agent SSM, abonnez-vous à la page Agent SSM Release Notes sur GitHub.

Installation manuelle de Agent SSM sur les instances Amazon Linux 2

Connectez-vous à votre instance Amazon Linux 2 et effectuez les opérations suivantes pour installer l'Agent SSM.

> **Note**
>
> Si vous utilisez une commande `yum` pour mettre à jour Agent SSM sur une instance gérée après l'installation ou la mise à niveau de l'agent à l'aide d'un document SSM`AWS-UpdateSSMAgent`, le message suivant pourrait s'afficher : « Avertissement : RPMDB a été modifiée hors de yum ». Ce message est attendu et peut être ignoré en toute sécurité.

Effectuez ces étapes sur chaque instance qui exécutera les commandes avec Systems Manager.

> **Important**
>
> Agent SSMest installé, par défaut, sur Amazon Linux une base AMIs de base datée de 2017.09 et ultérieures. Agent SSM est également installé, par défaut, sur les Amazon Linux 2 AMI. Vous devez installer manuellement Agent SSM sur les autres versions de Linux.

Pour installer l'Agent SSM sur Amazon Linux 2

Utilisez l'une des commandes suivantes pour télécharger et exécuter le programme d'installation de l'Agent SSM.

`région` représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de *`region`* prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

Intel 64-bit (x86_64)

```
sudo yum install -y https://s3.region.amazonaws.com/amazon-ssm-region/latest/linux_amd64/
amazon-ssm-agent.rpm
```

Intel 32-bit (x86)

```
sudo yum install -y https://s3.region.amazonaws.com/amazon-ssm-region/latest/linux_386/
amazon-ssm-agent.rpm
```

ARM 64-bit (arm64)

```
sudo yum install -y https://s3.region.amazonaws.com/amazon-ssm-region/latest/linux_arm64/
amazon-ssm-agent.rpm
```

> **Note**
>
> Si vous ne parvenez pas à télécharger l'agent à partir de la région que vous spécifiez, utilisez l'une des URL globales ci-dessous. Notez que même si les URL suivantes indiquent « ec2-downloads-windows », ce sont les URL correctes pour les systèmes d'exploitation Linux.
>
> - Intel 64-bit (x86_64)
>
> ```
> https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/linux_amd64/
> amazon-ssm-agent.rpm
> ```
>
> - Intel 32-bit (x86)
>
> ```
> https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/linux_386/amazon-
> ssm-agent.rpm
> ```
>
> - ARM 64-bit (arm64)
>
> ```
> https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/linux_arm64/
> amazon-ssm-agent.rpm
> ```

Après avoir installé l'agent, exécutez la commande suivante pour déterminer si l' Agent SSM est en cours d'exécution. La commande doit renvoyer le message `amazon-ssm-agent is running`.

```
sudo systemctl status amazon-ssm-agent
```

Si la commande précédente renvoie le message `amazon-ssm-agent is stopped` , exécutez les commandes suivantes :

```
sudo systemctl enable amazon-ssm-agent
sudo systemctl start amazon-ssm-agent
sudo systemctl status amazon-ssm-agent
```

> **Important**
>
> Une version mise à jour de l'Agent SSM est publiée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Si une version antérieure de l'agent est en cours d'exécution sur une instance, certains processus de l'Agent SSM peuvent échouer. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir l'Agent SSM à jour sur vos instances. Pour de plus amples informations, veuillez consulter Automatisation des mises à jour versAgent SSM (p. 109). Pour être alerté des mises à jour de Agent SSM, abonnez-vous à la page Agent SSM Release Notes sur GitHub.

Installation manuelle de Agent SSM sur les instances CentOS

Connectez-vous à votre CentOS instance et effectuez les opérations suivantes pour installer l' Agent SSM .

> **Note**
>
> Si vous utilisez une commande `yum` pour mettre à jour Agent SSM sur une instance gérée après l'installation ou la mise à niveau de l'agent à l'aide d'un document SSM`AWS-UpdateSSMAgent`, le

message suivant pourrait s'afficher : « Avertissement : RPMDB a été modifiée hors de yum ». Ce message est attendu et peut être ignoré en toute sécurité.

Effectuez ces étapes sur chaque instance qui exécutera les commandes avec Systems Manager.

CentOS 8.x

Pour installer Agent SSM sur CentOS 8.x

1. Assurez-vous que Python 2 ou Python 3 est installé sur votre instance CentOS 8. Ceci est nécessaire pour que Agent SSM fonctionne correctement.
2. Utilisez l'une des commandes suivantes pour télécharger et exécuter le programme d'installation de l'Agent SSM.

   région représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de *region* prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

   Intel 64-bit (x86_64)Instances :

   ```
   sudo dnf install -y https://s3.region.amazonaws.com/amazon-ssm-region/latest/
   linux_amd64/amazon-ssm-agent.rpm
   ```

   Intel 32-bit (x86)Instances :

   ```
   sudo dnf install -y https://s3.region.amazonaws.com/amazon-ssm-region/latest/
   linux_386/amazon-ssm-agent.rpm
   ```
3. Exécutez l'une des commandes suivantes afin de déterminer si l'Agent SSM est en cours d'exécution. La commande doit renvoyer le message `amazon-ssm-agent is running`.

   ```
   sudo systemctl status amazon-ssm-agent
   ```
4. Exécutez les commandes suivantes si la commande précédente a renvoyé `amazon-ssm-agent is stopped`.

   a. Lancez le service.

   ```
   sudo systemctl enable amazon-ssm-agent
   ```

   ```
   sudo systemctl start amazon-ssm-agent
   ```

   b. Vérifiez le statut de l'agent.

   ```
   sudo systemctl status amazon-ssm-agent
   ```

CentOS 7.x

Installer l'Agent SSM sur CentOS 7.x

1. Utilisez l'une des commandes suivantes pour télécharger et exécuter le programme d'installation de l'Agent SSM.

   région représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de

region prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

Intel 64-bit (x86_64)Instances :

```
sudo yum install -y https://s3.region.amazonaws.com/amazon-ssm-region/latest/
linux_amd64/amazon-ssm-agent.rpm
```

Intel 32-bit (x86)Instances :

```
sudo yum install -y https://s3.region.amazonaws.com/amazon-ssm-region/latest/
linux_386/amazon-ssm-agent.rpm
```

ARM 64-bit (arm64)Instances :

```
sudo yum install -y https://s3.region.amazonaws.com/amazon-ssm-region/latest/
linux_arm64/amazon-ssm-agent.rpm
```

2. Exécutez l'une des commandes suivantes afin de déterminer si l'Agent SSM est en cours d'exécution. La commande doit renvoyer le message `amazon-ssm-agent is running.`.

```
sudo systemctl status amazon-ssm-agent
```

3. Exécutez les commandes suivantes si la commande précédente a renvoyé `amazon-ssm-agent is stopped`.

   a. Lancez le service.

   ```
   sudo systemctl enable amazon-ssm-agent
   ```

   ```
   sudo systemctl start amazon-ssm-agent
   ```

   b. Vérifiez le statut de l'agent.

   ```
   sudo systemctl status amazon-ssm-agent
   ```

CentOS 6.x

Installer l’Agent SSM sur CentOS 6.x

1. Utilisez l'une des commandes suivantes pour télécharger et exécuter le programme d'installation de l'Agent SSM.

   région représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de *region* prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

   Intel 64-bit (x86_64)Instances :

   ```
   sudo yum install -y https://s3.region.amazonaws.com/amazon-ssm-region/latest/
   linux_amd64/amazon-ssm-agent.rpm
   ```

   Intel 32-bit (x86)Instances :

```
sudo yum install -y https://s3.region.amazonaws.com/amazon-ssm-region/latest/
linux_386/amazon-ssm-agent.rpm
```

2. Exécutez l'une des commandes suivantes afin de déterminer si l'Agent SSM est en cours d'exécution. La commande doit renvoyer le message `amazon-ssm-agent is running.`.

```
sudo status amazon-ssm-agent
```

3. Exécutez les commandes suivantes si la commande précédente a renvoyé `amazon-ssm-agent is stopped`.

   a. Lancez le service.

   ```
   sudo start amazon-ssm-agent
   ```

   b. Vérifiez le statut de l'agent.

   ```
   sudo status amazon-ssm-agent
   ```

Note

Si vous ne parvenez pas à télécharger l'agent à partir de la région que vous spécifiez, utilisez l'une des URL globales ci-dessous. Notez que même si les URL suivantes indiquent « ec2-downloads-windows », ce sont les URL correctes pour les systèmes d'exploitation Linux.

- Intel 64-bit (x86_64)

```
https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/linux_amd64/
amazon-ssm-agent.rpm
```

- Intel 32-bit (x86)

```
https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/linux_386/amazon-
ssm-agent.rpm
```

- ARM 64-bit (arm64)

```
https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/linux_arm64/
amazon-ssm-agent.rpm
```

Important

Une version mise à jour de l'Agent SSM est publiée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Si une version antérieure de l'agent est en cours d'exécution sur une instance, certains processus de l'Agent SSM peuvent échouer. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir l'Agent SSM à jour sur vos instances. Pour de plus amples informations, veuillez consulter Automatisation des mises à jour versAgent SSM (p. 109). Pour être alerté des mises à jour de Agent SSM, abonnez-vous à la page Agent SSM Release Notes sur GitHub.

Installation manuelle de Agent SSM sur les instances Debian Server

Connectez-vous à votre instance Debian Server et effectuez les étapes de l'une des procédures suivantes pour installer Agent SSM sur chaque instance qui exécutera les commandes à l'aide de Systems Manager.

Installer l'Agent SSM sur des instances Debian Server

Debian Server 9 and 10 64-bit (deb)

Pour installer l'Agent SSM sur les instances Debian Server 9 64 bits (avec un package d'installation deb)

1. Connectez-vous à votre instance Debian Server et effectuez les opérations suivantes pour installer l'Agent SSM. Effectuez ces étapes sur chaque instance qui exécutera les commandes avec Systems Manager.

   Créez un répertoire temporaire sur l'instance.

   ```
   mkdir /tmp/ssm
   ```

   Passez au répertoire temporaire.

   ```
   cd /tmp/ssm
   ```

   Exécutez les commandes suivantes.

   > **Note**
   >
   > Même si l'URL de téléchargement suivante indique « ec2-downloads-windows », il s'agit de l'URL correcte.

   ```
   wget https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/debian_amd64/
   amazon-ssm-agent.deb
   ```

   ```
   sudo dpkg -i amazon-ssm-agent.deb
   ```

2. Exécutez la commande suivante afin de déterminer si l'Agent SSM est en cours d'exécution.

   ```
   sudo systemctl status amazon-ssm-agent
   ```

3. Exécutez la commande suivante pour démarrer le service si la commande précédente a renvoyé `amazon-ssm-agent is stopped`, `inactive` ou `disabled`.

   ```
   sudo systemctl enable amazon-ssm-agent
   ```

4. Utilisez la commande suivante pour vérifier le statut de l'agent :

   ```
   sudo systemctl status amazon-ssm-agent
   ```

Debian Server 8 64-bit (deb)

Installer l'Agent SSM sur les instances Debian Server 8 64 bits (avec un package d'installation deb)

1. Connectez-vous à votre instance Debian Server et effectuez les opérations suivantes pour installer l'Agent SSM. Effectuez ces étapes sur chaque instance qui exécutera les commandes avec Systems Manager.

   Créez un répertoire temporaire sur l'instance.

   ```
   mkdir /tmp/ssm
   ```

   Passez au répertoire temporaire.

   ```
   cd /tmp/ssm
   ```

   Exécutez les commandes suivantes.

   > **Note**
   >
   > Même si l'URL de téléchargement suivante indique « ec2-downloads-windows », il s'agit de l'URL correcte.

   ```
   wget https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/debian_amd64/
   amazon-ssm-agent.deb
   ```

   ```
   sudo dpkg -i amazon-ssm-agent.deb
   ```

2. Exécutez la commande suivante afin de déterminer si l'Agent SSM est en cours d'exécution.

   ```
   sudo systemctl status amazon-ssm-agent
   ```

3. Exécutez la commande suivante pour démarrer le service si la commande précédente a renvoyé `amazon-ssm-agent is stopped`, `inactive` ou `disabled`.

   ```
   sudo systemctl enable amazon-ssm-agent
   ```

4. Utilisez la commande suivante pour vérifier le statut de l'agent :

   ```
   sudo systemctl status amazon-ssm-agent
   ```

> **Important**
>
> Une version mise à jour de l'Agent SSM est publiée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Si une version antérieure de l'agent est en cours d'exécution sur une instance, certains processus de l'Agent SSM peuvent échouer. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir l'Agent SSM à jour sur vos instances. Pour de plus amples informations, veuillez consulter Automatisation des mises à jour versAgent SSM (p. 109). Pour être alerté des mises à jour de Agent SSM, abonnez-vous à la page Agent SSM Release Notes sur GitHub.

Installation manuelle de Agent SSM sur les instances Oracle Linux

Connectez-vous à votre instance Oracle Linux et effectuez les opérations suivantes pour installer l'Agent SSM.

> **Note**
>
> Si vous utilisez une commande `yum` pour mettre à jour Agent SSM sur une instance gérée après l'installation ou la mise à niveau de l'agent à l'aide d'un document SSM`AWS-UpdateSSMAgent`, le message suivant pourrait s'afficher : « Avertissement : RPMDB a été modifiée hors de yum ». Ce message est attendu et peut être ignoré en toute sécurité.

Effectuez ces étapes sur chaque instance qui exécutera les commandes avec Systems Manager.

Pour installer l'Agent SSM sur Oracle Linux

Utilisez l'une des commandes suivantes pour télécharger et exécuter le programme d'installation de l'Agent SSM.

`région` représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de *`region`* prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

Intel 64-bit (x86_64)

```
sudo yum install -y https://s3.region.amazonaws.com/amazon-ssm-region/latest/linux_amd64/
amazon-ssm-agent.rpm
```

> **Note**
>
> Si vous ne parvenez pas à télécharger l'agent à partir de la région que vous spécifiez, utilisez l'une des URL globales ci-dessous. Notez que même si les URL suivantes indiquent « ec2-downloads-windows », ce sont les URL correctes pour les systèmes d'exploitation Linux.
> Intel 64-bit (x86_64)
>
> ```
> https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/linux_amd64/amazon-
> ssm-agent.rpm
> ```

Après avoir installé l'agent, exécutez la commande suivante pour déterminer si l' Agent SSM est en cours d'exécution. La commande doit renvoyer le message `amazon-ssm-agent is running`.

```
sudo systemctl status amazon-ssm-agent
```

Si la commande précédente renvoie le message `amazon-ssm-agent is stopped` , exécutez les commandes suivantes :

```
sudo systemctl enable amazon-ssm-agent
sudo systemctl start amazon-ssm-agent
sudo systemctl status amazon-ssm-agent
```

> **Important**
>
> Une version mise à jour de l'Agent SSM est publiée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Si une version antérieure de l'agent est en cours d'exécution sur une instance, certains processus de l'Agent SSM peuvent échouer. C'est pourquoi nous vous recommandons

d'automatiser le processus permettant de maintenir l'Agent SSM à jour sur vos instances. Pour de plus amples informations, veuillez consulter Automatisation des mises à jour versAgent SSM (p. 109). Pour être alerté des mises à jour de Agent SSM, abonnez-vous à la page Agent SSM Release Notes sur GitHub.

Installation manuelle de Agent SSM sur les instances Red Hat Enterprise Linux

Connectez-vous à votre instance RHEL et effectuez les opérations suivantes pour installer l'Agent SSM.

> **Note**
>
> Si vous utilisez une commande `yum` pour mettre à jour Agent SSM sur une instance gérée après l'installation ou la mise à niveau de l'agent à l'aide d'un document SSM`AWS-UpdateSSMAgent`, le message suivant pourrait s'afficher : « Avertissement : RPMDB a été modifiée hors de yum ». Ce message est attendu et peut être ignoré en toute sécurité.

Effectuez ces étapes sur chaque instance qui exécutera les commandes avec Systems Manager.

RHEL 8.x

Pour installer Agent SSM sur Red Hat Enterprise Linux 8.x

1. Assurez-vous que Python 2 ou Python 3 est installé sur votre instance RHEL 8. Ceci est nécessaire pour que Agent SSM fonctionne correctement.
2. Utilisez l'une des commandes suivantes pour télécharger et exécuter le programme d'installation de l'Agent SSM.

   `région` représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de *`region`* prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

   Intel 64-bit (x86_64)Instances :

   ```
   sudo dnf install -y https://s3.region.amazonaws.com/amazon-ssm-region/latest/
   linux_amd64/amazon-ssm-agent.rpm
   ```

   Intel 32-bit (x86)Instances :

   ```
   sudo dnf install -y https://s3.region.amazonaws.com/amazon-ssm-region/latest/
   linux_386/amazon-ssm-agent.rpm
   ```

   ARM 64-bit (arm64)Instances :

   ```
   sudo dnf install -y https://s3.region.amazonaws.com/amazon-ssm-region/latest/
   linux_arm64/amazon-ssm-agent.rpm
   ```

3. Exécutez l'une des commandes suivantes afin de déterminer si l'Agent SSM est en cours d'exécution. La commande doit renvoyer le message `amazon-ssm-agent is running`.

   ```
   sudo systemctl status amazon-ssm-agent
   ```

4. Exécutez les commandes suivantes si la commande précédente a renvoyé `amazon-ssm-agent is stopped`.

   a. Lancez le service.

```
sudo systemctl enable amazon-ssm-agent
```

```
sudo systemctl start amazon-ssm-agent
```

b. Vérifiez le statut de l'agent.

```
sudo systemctl status amazon-ssm-agent
```

RHEL 7.x

Installer l'Agent SSM sur Red Hat Enterprise Linux 7.x

1. Utilisez l'une des commandes suivantes pour télécharger et exécuter le programme d'installation de l'Agent SSM.

   région représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de *region* prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

   Intel 64-bit (x86_64)Instances :

   ```
   sudo yum install -y https://s3.region.amazonaws.com/amazon-ssm-region/latest/
   linux_amd64/amazon-ssm-agent.rpm
   ```

   Intel 32-bit (x86)Instances :

   ```
   sudo yum install -y https://s3.region.amazonaws.com/amazon-ssm-region/latest/
   linux_386/amazon-ssm-agent.rpm
   ```

   ARM 64-bit (arm64)Instances :

   ```
   sudo yum install -y https://s3.region.amazonaws.com/amazon-ssm-region/latest/
   linux_arm64/amazon-ssm-agent.rpm
   ```

2. Exécutez l'une des commandes suivantes afin de déterminer si l'Agent SSM est en cours d'exécution. La commande doit renvoyer le message `amazon-ssm-agent is running`.

   ```
   sudo systemctl status amazon-ssm-agent
   ```

3. Exécutez les commandes suivantes si la commande précédente a renvoyé `amazon-ssm-agent is stopped`.

   a. Lancez le service.

   ```
   sudo systemctl enable amazon-ssm-agent
   ```

   ```
   sudo systemctl start amazon-ssm-agent
   ```

   b. Vérifiez le statut de l'agent.

   ```
   sudo systemctl status amazon-ssm-agent
   ```

RHEL 6.x

Installer l'Agent SSM sur Red Hat Enterprise Linux 6.x

1. Utilisez l'une des commandes suivantes pour télécharger et exécuter le programme d'installation de l'Agent SSM.

   région représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de *region* prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

   Intel 64-bit (x86_64)Instances :

   ```
   sudo yum install -y https://s3.region.amazonaws.com/amazon-ssm-region/latest/
   linux_amd64/amazon-ssm-agent.rpm
   ```

   Intel 32-bit (x86)Instances :

   ```
   sudo yum install -y https://s3.region.amazonaws.com/amazon-ssm-region/latest/
   linux_386/amazon-ssm-agent.rpm
   ```

2. Exécutez l'une des commandes suivantes afin de déterminer si l'Agent SSM est en cours d'exécution. La commande doit renvoyer le message `amazon-ssm-agent is running`.

   ```
   sudo status amazon-ssm-agent
   ```

3. Exécutez les commandes suivantes si la commande précédente a renvoyé `amazon-ssm-agent is stopped`.

   a. Lancez le service.

   ```
   sudo start amazon-ssm-agent
   ```

   b. Vérifiez le statut de l'agent.

   ```
   sudo status amazon-ssm-agent
   ```

Note

Si vous ne parvenez pas à télécharger l'agent à partir de la région que vous spécifiez, utilisez l'une des URL globales ci-dessous. Notez que même si les URL suivantes indiquent « ec2-downloads-windows », ce sont les URL correctes pour les systèmes d'exploitation Linux.

- Intel 64-bit (x86_64)

  ```
  https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/linux_amd64/
  amazon-ssm-agent.rpm
  ```

- Intel 32-bit (x86)

  ```
  https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/linux_386/amazon-
  ssm-agent.rpm
  ```

- ARM 64-bit (arm64)

```
https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/linux_arm64/
amazon-ssm-agent.rpm
```

Important

Une version mise à jour de l'Agent SSM est publiée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Si une version antérieure de l'agent est en cours d'exécution sur une instance, certains processus de l'Agent SSM peuvent échouer. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir l'Agent SSM à jour sur vos instances. Pour de plus amples informations, veuillez consulter Automatisation des mises à jour versAgent SSM (p. 109). Pour être alerté des mises à jour de Agent SSM, abonnez-vous à la page Agent SSM Release Notes sur GitHub.

Installation manuelle de Agent SSM sur les instances SUSE Linux Enterprise Server 12

Connectez-vous à votre instance SLES et effectuez les opérations suivantes pour installer l'Agent SSM. Effectuez ces étapes sur chaque instance qui exécutera les commandes avec Systems Manager.

Pour installer l'Agent SSM sur SUSE Linux Enterprise Server

1. Créez un répertoire temporaire sur l'instance.

```
mkdir /tmp/ssm
```

2. Passez au répertoire temporaire.

```
cd /tmp/ssm
```

3. Exécutez les commandes suivantes l'une après l'autre pour télécharger et exécuter le programme d'installation de l'Agent SSM.

   Note

   Même si l'URL de téléchargement suivante indique « ec2-downloads-windows », il s'agit de l'URL correcte.

   Instances 64 bits :

```
wget https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/linux_amd64/amazon-
ssm-agent.rpm
```

```
sudo rpm --install amazon-ssm-agent.rpm
```

4. Exécutez la commande suivante afin de déterminer si l'Agent SSM est en cours d'exécution. La commande doit renvoyer le message `amazon-ssm-agent is running`.

```
sudo systemctl status amazon-ssm-agent
```

5. Exécutez les commandes suivantes si la commande précédente renvoie le message `amazon-ssm-agent is stopped`.

   a. Lancez le service.

```
sudo systemctl enable amazon-ssm-agent
```

```
sudo systemctl start amazon-ssm-agent
```

b. Vérifiez le statut de l'agent.

```
sudo systemctl status amazon-ssm-agent
```

Important

Une version mise à jour de l'Agent SSM est publiée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Si une version antérieure de l'agent est en cours d'exécution sur une instance, certains processus de l'Agent SSM peuvent échouer. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir l'Agent SSM à jour sur vos instances. Pour de plus amples informations, veuillez consulter Automatisation des mises à jour versAgent SSM (p. 109). Pour être alerté des mises à jour de Agent SSM, abonnez-vous à la page Agent SSM Release Notes sur GitHub.

Installation manuelle de Agent SSM sur les instances Ubuntu Server

Connectez-vous à votre instance Ubuntu Server et effectuez les étapes de l'une des procédures suivantes pour installer Agent SSM sur chaque instance qui exécutera les commandes à l'aide de Systems Manager.

A propos de l'installation de l'Agent SSM sur les instances Ubuntu Server 16.04 64 bits

À partir des instances créées à partir de Ubuntu Server la version 16.04 AMIs identifiée par `20180627` , Agent SSM est préinstallé à l'aide de packages Snap. Exemple : `ubuntu/images/hvm-ssd/ubuntu-xenial-16.04-amd64-server-20180627`. Pour les instances créées à partir d' précédemment, vous devez continuer à utiliser les packages d'installation deb.AMIs

Pour les instances créées à partir d'un Ubuntu Server 16.04 64 bits, veillez à suivre la procédure correcte pour votre type AMI d'installation :Agent SSM

- Instances créées à partir dAMIs' avec un identifiant **`20180627`** ou version ultérieure :Installer l'Agent SSM sur des instances Ubuntu Server (p. 92)
- Instances créées à partir dAMIs' antérieures à **`20180627`** :Installer l'Agent SSM sur des instances Ubuntu Server (p. 92)

Important

Sachez que si une instance comporte plusieurs installations de l' Agent SSM (par exemple, une effectuée à l'aide d'un snap et une installée à l'aide d'un programme d'installation deb), vos opérations d'agent ne fonctionneront pas correctement.

Vous pouvez vérifier la date de création AMI de l'ID source d'une instance à l'aide de l'une des méthodes suivantes. Notez que ces procédures s'appliquent uniquement à AWS géré AMIs par .

Vérification de la date de création dAMI'un ID source (console)

1. Ouvrez la console Amazon EC2 à l'adresse https://console.aws.amazon.com/ec2/.

2. Dans le panneau de navigation de gauche, choisissez Instances.
3. Sélectionnez une instance.
4. Dans l'onglet Description, recherchez un `YYYYMMDD` identifiant dans la valeur du champ ID d'AMI. Exemple : `ubuntu/images/hvm-ssd/ubuntu-xenial-16.04-amd64-server-20180627`.

Vérification de la date de création dAMI'un ID source (AWS CLI)

- Exécutez la commande suivante .

```
aws ec2 describe-images --image-id ami-id
```

`ami-id` représente l'ID d'un AMI fourni par AWS , tel que `ami-07c8bc5c1ce9598c3` .

En cas de réussite, la commande renvoie des informations telles que les suivantes, dans lesquelles vous pouvez vérifier les `CreationDate` champs `Name` et pour obtenir des informations.

```
{
    "Images": [
        {
            "Architecture": "x86_64",
            "CreationDate": "2020-07-24T20:40:27.000Z",
            "ImageId": "ami-07c8bc5c1ce9598c3",
-- truncated --
            "ImageOwnerAlias": "amazon",
            "Name": "amzn2-ami-hvm-2.0.20200722.0-x86_64-gp2",
            "RootDeviceName": "/dev/xvda",
            "RootDeviceType": "ebs",
            "SriovNetSupport": "simple",
            "VirtualizationType": "hvm"
        }
    ]
}
```

Installer l’Agent SSM sur des instances Ubuntu Server

Ubuntu Server 20.10 STR & 20.04, 18.04, and 16.04 LTS 64-bit (Snap)

Pour installer Agent SSM sur des instances Ubuntu Server 20.10 STR & 20.04, 18.04 et 16.04 LTS 64 bits (avec un package Snap)

1. Agent SSML' est installé par défaut sur Ubuntu Server 20.04, 18.04 et 16.04 LTS 64 bits AMIs avec un identifiant de `20180627` ou ultérieur.

   Vous pouvez utiliser le script suivant si vous avez besoin d'installer l'Agent SSM sur un serveur sur site ou que vous devez réinstaller l'agent. Vous n'avez pas besoin de spécifier une URL pour le téléchargement, car la commande `snap` télécharge automatiquement l'agent à partir de la boutique d'applications Snap à l'adresse https://snapcraft.io.

   ```
   sudo snap install amazon-ssm-agent --classic
   ```

   Important

   Le canal candidat dans le magasin Snap contient la dernière version de Agent SSM ; pas le canal stable. Si vous souhaitez suivre les informations de Agent SSM version sur le canal candidat, exécutez la commande suivante sur vos instances Ubuntu Server STR 20.10 et 20.04, 18.04 et 16.04 LTS 64 bits.

```
sudo snap switch --channel=candidate amazon-ssm-agent
```

Note

Notez les détails suivants à propos de Agent SSM sur Ubuntu Server 20.10 STR & 20.04, 18.04 et 16.04 LTS :

- En raison d'un problème connu lié à Snap, vous pouvez voir une erreur `Maximum timeout exceeded` s'afficher lors de l'exécution des commandes `snap`. Si vous recevez cette erreur, exécutez les commandes suivantes une par une pour démarrer l'agent, l'arrêter et vérifier son statut :

```
sudo systemctl start snap.amazon-ssm-agent.amazon-ssm-agent.service
```

```
sudo systemctl stop snap.amazon-ssm-agent.amazon-ssm-agent.service
```

```
sudo systemctl status snap.amazon-ssm-agent.amazon-ssm-agent.service
```

- Les fichiers Ubuntu Server du Agent SSM programme d'installation des sur STR 20.10 & 20.04, 18.04 et 16.04 LTS, y compris les fichiers binaires et les fichiers de configuration de l'agent, sont stockés dans le répertoire suivant : `/snap/amazon-ssm-agent/current/`. Si vous apportez des modifications aux fichiers de configuration de ce répertoire, vous devez copier ces fichiers du dossier vers le `/snap` dossier.`/etc/amazon/ssm/` Les fichiers journaux et de bibliothèque n'ont pas changé (`/var/lib/amazon/ssm`, `/var/log/amazon/ssm`).
- Sur Ubuntu Server les versions 20.10 STR et 20.04 et 18.04 LTS, utilisez Snaps uniquement. N'installez pas de packages deb. Vérifiez également qu'une seule instance de l'agent est installée et en cours d'exécution sur vos instances.
- Sur Ubuntu Server 20.10 STR & 20.04, 18.04 et 16.04 LTS, Agent SSM prend en charge l'architecture de processeur arm64.
- Sur Ubuntu Server 16.04, Agent SSM est installé à l'aide de Snaps ou de packages d'installation deb, en fonction de la version de la version 16.04AMI.

2. Exécutez la commande suivante afin de déterminer si l'Agent SSM est en cours d'exécution.

```
sudo snap list amazon-ssm-agent
```

3. Exécutez la commande suivante pour démarrer le service si la commande précédente a renvoyé `amazon-ssm-agent is stopped`, `inactive` ou `disabled`.

```
sudo snap start amazon-ssm-agent
```

4. Vérifiez le statut de l'agent.

```
sudo snap services amazon-ssm-agent
```

Ubuntu Server 16.04 and 14.04 64-bit (deb)

Installer l’Agent SSM sur des instances Ubuntu Server 16.04 et 14.04 64 bits (avec un package d'installation deb)

1. Vous pouvez utiliser le script suivant si vous avez besoin d'installer l'Agent SSM sur un serveur sur site ou que vous devez réinstaller l'agent.

> **Important**
>
> Agent SSMest installé par défaut sur les instances créées à partir de Ubuntu Server 16.04 LTS 64 bits AMIs avec un identifiant ou ultérieur.`20180627` Les instances créées à partir d' AMIs avec des identifiants antérieurs, par exemple `20171121.1` et `20180522` , doivent continuer à utiliser les programmes d'installation deb.
> Si l'Agent SSM est installé sur votre instance avec un Snap et que vous installez ou mettez à jour l'Agent SSM à l'aide d'un package d'installation deb, l'installation ou les opérations de l'Agent SSM peuvent échouer.

Créez un répertoire temporaire sur l'instance.

```
mkdir /tmp/ssm
```

Passez au répertoire temporaire.

```
cd /tmp/ssm
```

Exécutez les commandes suivantes.

`région` représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de *`region`* prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

```
wget https://s3.region.amazonaws.com/amazon-ssm-region/latest/debian_amd64/amazon-
ssm-agent.deb
```

```
sudo dpkg -i amazon-ssm-agent.deb
```

2. Exécutez l'une des commandes suivantes afin de déterminer si l'Agent SSM est en cours d'exécution.

   Ubuntu Server 16.04 :

   ```
   sudo systemctl status amazon-ssm-agent
   ```

   Ubuntu Server 14.04 :

   ```
   sudo status amazon-ssm-agent
   ```

3. Exécutez l'une des commandes suivantes pour démarrer le service si la commande précédente a renvoyé `amazon-ssm-agent is stopped`, `inactive` ou `disabled`.

   Ubuntu Server 16.04 :

   ```
   sudo systemctl enable amazon-ssm-agent
   ```

   Ubuntu Server 14.04 :

   ```
   sudo start amazon-ssm-agent
   ```

4. Exécutez l'une des commandes suivantes pour vérifier le statut de l'agent :

Ubuntu Server 16.04 :

```
sudo systemctl status amazon-ssm-agent
```

Ubuntu Server 14.04 :

```
sudo status amazon-ssm-agent
```

Ubuntu Server 16.04 and 14.04 32-bit

Installer l'Agent SSM sur des instances Ubuntu Server 16.04 et 14.04 32 bits

1. Créez un répertoire temporaire sur l'instance.

   ```
   mkdir /tmp/ssm
   ```

   Passez au répertoire temporaire.

   ```
   cd /tmp/ssm
   ```

   Exécutez les commandes suivantes.

   région représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de *region* prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

   ```
   wget https://s3.region.amazonaws.com/amazon-ssm-region/latest/debian_386/amazon-
   ssm-agent.deb
   ```

   ```
   sudo dpkg -i amazon-ssm-agent.deb
   ```

2. Exécutez la commande suivante afin de déterminer si l'Agent SSM est en cours d'exécution:

   ```
   sudo status amazon-ssm-agent
   ```

3. Exécutez les commandes suivantes si la commande précédente a renvoyé `amazon-ssm-agent is stopped`, `inactive`, ou `disabled`.

   a. Démarrez l'agent:

   ```
   sudo start amazon-ssm-agent
   ```

   b. Vérifier le statut de l'agent :

   ```
   sudo status amazon-ssm-agent
   ```

Note

Si vous ne parvenez pas à télécharger l'agent à partir de la région que vous spécifiez, utilisez l'une des URL globales ci-dessous. Notez que même si les URL suivantes indiquent « ec2-downloads-windows », ce sont les URL correctes pour les systèmes d'exploitation Linux.

- Intel 64-bit (x86_64)

```
https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/debian_amd64/
amazon-ssm-agent.deb
```

- Intel 32-bit (x86)

```
https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/debian_386/amazon-
ssm-agent.deb
```

Important

Une version mise à jour de l'Agent SSM est publiée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Si une version antérieure de l'agent est en cours d'exécution sur une instance, certains processus de l'Agent SSM peuvent échouer. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir l'Agent SSM à jour sur vos instances. Pour de plus amples informations, veuillez consulter Automatisation des mises à jour versAgent SSM (p. 109). Pour être alerté des mises à jour de Agent SSM, abonnez-vous à la page Agent SSM Release Notes sur GitHub.

Configurer Agent SSM pour utiliser un proxy

Vous pouvez configurer Agent SSM pour communiquer via un proxy HTTP en créant un fichier de configuration de remplacement et en ajoutant `http_proxy` les `https_proxy` paramètres , et au fichier.`no_proxy` Un fichier de remplacement conserve également les paramètres de proxy si vous installez des versions plus récentes ou antérieures de l'Agent SSM. Cette section inclut les procédures de création d'un fichier de remplacement dans les environnements upstart et systemd.

Note

Les instances créées depuis une AMI Amazon Linux qui utilisent un proxy doivent exécuter une version actuelle du module Python `requests` afin de prendre en charge les opérations du Gestionnaire de correctifs. Pour de plus amples informations, veuillez consulter Mettre à niveau le module de demandes Python sur Amazon Linux les instances qui utilisent un serveur proxy (p. 98).

Rubriques

- Configurer l'Agent SSM afin d'utiliser un proxy (Upstart) (p. 96)
- Configuration de l'Agent SSM afin d'utiliser un proxy (systemd) (p. 97)
- Mettre à niveau le module de demandes Python sur Amazon Linux les instances qui utilisent un serveur proxy (p. 98)

Configurer l'Agent SSM afin d'utiliser un proxy (Upstart)

Utilisez la procédure suivante pour créer un fichier de configuration de remplacement pour un `upstart` environnement.

Pour configurer Agent SSM afin d'utiliser un proxy (upstart)

1. Connectez-vous à l'instance sur laquelle vous avez installé l'Agent SSM.
2. Ouvrez un éditeur simple comme VIM, et selon que vous utilisez un serveur proxy HTTP ou un serveur proxy HTTPS, ajoutez l'une des configurations suivantes.

Pour un serveur proxy HTTP :

```
env http_proxy=http://hostname:port
env https_proxy=http://hostname:port
env no_proxy=169.254.169.254
```

Pour un serveur proxy HTTPS :

```
env http_proxy=http://hostname:port
env https_proxy=https://hostname:port
env no_proxy=169.254.169.254
```

> **Note**
>
> Vous devez ajouter le paramètre `no_proxy` au fichier et spécifier l'adresse IP indiquée ici. Il s'agit du point de terminaison des métadonnées de l'instance pour Systems Manager. Sans cette adresse IP, les appels à Systems Manager échouent.

3. Enregistrez le fichier sous le nom `amazon-ssm-agent.override` à l'emplacement suivant :`/etc/init/`
4. Arrêtez et redémarrez l'Agent SSM à l'aide des commandes suivantes :

```
sudo stop amazon-ssm-agent
sudo start amazon-ssm-agent
```

> **Note**
>
> Pour plus d'informations sur l'utilisation des `.override` fichiers dans les environnements Upstart, consultez init : Configuration de la tâche du démon d'initialisation Upstart.

Configuration de l'Agent SSM afin d'utiliser un proxy (systemd)

Procédez comme suit Agent SSM pour configurer afin d'utiliser un proxy dans un `systemd` environnement.

> **Note**
>
> Certaines étapes de cette procédure contiennent des instructions explicites pour les Ubuntu Server instances sur lesquelles Agent SSM a été installé à l'aide de Snap.

1. Connectez-vous à l'instance sur laquelle vous avez installé l'Agent SSM.
2. Exécutez l'une des commandes suivantes, selon le type du système d'exploitation.
   - Sur Ubuntu Server les instances où Agent SSM est installé à l'aide d'un snap :

     ```
     systemctl edit snap.amazon-ssm-agent.amazon-ssm-agent
     ```

     Sur d'autres systèmes d'exploitation :

     ```
     systemctl edit amazon-ssm-agent
     ```
3. Ouvrez un éditeur simple comme VIM, et selon que vous utilisez un serveur proxy HTTP ou un serveur proxy HTTPS, ajoutez l'une des configurations suivantes.

   Pour un serveur proxy HTTP :

```
[Service]
Environment="http_proxy=http://hostname:port"
Environment="https_proxy=http://hostname:port"
Environment="no_proxy=169.254.169.254"
```

Pour un serveur proxy HTTPS :

```
[Service]
Environment="http_proxy=http://hostname:port"
Environment="https_proxy=https://hostname:port"
Environment="no_proxy=169.254.169.254"
```

> **Note**
>
> Vous devez ajouter le paramètre `no_proxy` au fichier et spécifier l'adresse IP indiquée ici. Il s'agit du point de terminaison des métadonnées de l'instance pour Systems Manager. Sans cette adresse IP, les appels à Systems Manager échouent.

4. Enregistrez vos modifications. Le système crée automatiquement l'un des fichiers suivants, selon le type du système d'exploitation.
   - Sur Ubuntu Server les instances où Agent SSM est installé à l'aide d'un snap :

     `etc/systemd/system/snap.amazon-ssm-agent.amazon-ssm-agent.service.d/`**`override.conf`**
   - Sur Amazon Linux 2 les instances :

     `etc/systemd/system/amazon-ssm-agent.service.d/`**`override.conf`**
   - Sur d'autres systèmes d'exploitation :

     `etc/systemd/system/amazon-ssm-agent.service.d/`**`amazon-ssm-agent.override`**
5. Redémarrez en utilisant l'une des commandes suivantes, selon le type de système d'exploitation.Agent SSM
   - Sur Ubuntu Server les instances installées à l'aide d'un snap :

     ```
     systemctl start snap.amazon-ssm-agent.amazon-ssm-agent
     ```
   - Sur d'autres systèmes d'exploitation :

     ```
     sudo systemctl stop amazon-ssm-agent ; sudo systemctl daemon-reload
     ```

> **Note**
>
> Pour plus d'informations sur l'utilisation des `.override` fichiers dans les environnements systemd, consultez Modification des fichiers d'unité existants dans le Guide de l'administrateur système Red Hat Enterprise Linux 7.

Mettre à niveau le module de demandes Python sur Amazon Linux les instances qui utilisent un serveur proxy

Pour corriger une instance qui utilise un proxy et qui a été créée à partir d'un Amazon Linux AMI , Gestionnaire de correctifs nécessite qu'une version récente du `requests` module Python soit installée sur l'instance. Nous recommandons une mise à niveau systématique avec la version la plus récente.

Pour vous assurer que la dernière version du `requests` module Python est installée, procédez comme suit :

1. Connectez-vous à l'instance Amazon Linux ou utilisez le document SSM AWS-RunShellScript dans Fonctionnalité Exécuter la commande, puis exécutez la commande suivante sur l'instance :

   ```
   pip list | grep requests
   ```

   - Si le module est installé, la demande renvoie le numéro de version dans une réponse similaire à ce qui suit :

     ```
     requests (1.2.3)
     ```

   - Si le module n'a pas encore été installé, exécutez la commande suivante pour l'installer :

     ```
     pip install requests
     ```

   - Si pip lui-même n'a pas encore été installé, exécutez la commande suivante pour l'installer :

     ```
     sudo yum install -y python-pip
     ```

2. Si le module est installé, mais que la version répertoriée est antérieure à `2.18.4` (comme `1.2.3` illustré à l'étape précédente), exécutez la commande suivante pour effectuer la mise à niveau vers la dernière version de l' Python Module `requests` :

   ```
   pip install requests --upgrade
   ```

Désinstaller Agent SSM sur les instances Linux

Utilisez les commandes suivantes pour désinstaller l'Agent SSM.

Amazon Linux, Amazon Linux 2 , RHEL et CentOS

```
sudo yum erase amazon-ssm-agent -y
```

Ubuntu Server

- installations des packages deb

  ```
  sudo dpkg -r amazon-ssm-agent
  ```

- installations des packages snap

  ```
  sudo snap remove amazon-ssm-agent
  ```

Debian Server

```
sudo dpkg -r amazon-ssm-agent
```

SLES

```
sudo rpm --erase amazon-ssm-agent
```

Installation et configuration Agent SSM de sur les instances EC2 pourWindows Server

Agent SSMest installé par défaut sur les éléments suivants Amazon Machine Images (AMIs) :

- AMIs Windows Server 2008-2012 R2 publiées à partir de novembre 2016
- Windows Server 2016 et 2019

Windows ServerAMIspubliée avant novembre 2016 utilisez le EC2Config service pour traiter les demandes et configurer les instances.

A moins d'avoir une raison précise d'utiliser le EC2Config service , ou une version antérieure d' Agent SSM , pour traiter Systems Manager les demandes, nous vous recommandons de télécharger et d'installer la dernière version d' Agent SSM pour chacune de vos instances EC2 ou des instances hybrides configurées pour Systems Manager .

> **Important**
>
> Une version mise à jour de l'Agent SSM est publiée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Si une version antérieure de l'agent est en cours d'exécution sur une instance, certains processus de l'Agent SSM peuvent échouer. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir l'Agent SSM à jour sur vos instances. Pour de plus amples informations, veuillez consulter Automatisation des mises à jour versAgent SSM (p. 109). Pour être alerté des mises à jour de Agent SSM, abonnez-vous à la page Agent SSM Release Notes sur GitHub.

Pour afficher des informations sur les différentes versions de Agent SSM, consultez les notes de mise à jour.

Rubriques

- Installation manuelle Agent SSM sur les instances EC2 pourWindows Server (p. 100)
- Configurer l'Agent SSM pour utiliser un proxy pour les instances Windows Server (p. 102)

Installation manuelle Agent SSM sur les instances EC2 pourWindows Server

Agent SSMest installé par défaut sur les éléments suivants Amazon Machine Images (AMIs) :

- AMIs Windows Server 2008-2012 R2 publiées à partir de novembre 2016
- Windows Server 2016 et 2019

Si votre instance est une instance Windows Server 2008-2012 R2 créée avant novembre 2016, EC2Config traite les demandes Systems Manager sur votre instance. Nous vous recommandons de mettre à niveau vos instances existantes pour utiliser la dernière version d' EC2Config . En utilisant le dernier EC2Config programme d'installation, vous installez Agent SSM côte à côte avec EC2Config. Cette version côte à côte d' Agent SSM est compatible avec vos instances créées à partir d' précédentes Windows Server AMIs et vous permet d'utiliser SSM les fonctions publiées après novembre 2016. Pour plus d'informations sur l'installation de la dernière version du EC2Config service, consultez Installation de la dernière version d'EC2Config dans le Amazon EC2 Guide de l'utilisateur pour les instances Windows .

> **Important**
>
> Une version mise à jour de l'Agent SSM est publiée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Si une version antérieure de l'agent est en cours d'exécution sur une instance, certains processus de l'Agent SSM peuvent échouer. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir l'Agent SSM à jour sur vos instances. Pour de plus amples informations, veuillez consulter Automatisation des mises à jour versAgent SSM (p. 109). Pour être alerté des mises à jour de Agent SSM, abonnez-vous à la page Agent SSM Release Notes sur GitHub.

Si nécessaire, vous pouvez aussi télécharger et installer manuellement la version la plus récente de l'Agent SSM sur votre instance EC2 pour Windows Server à l'aide de la procédure suivante.

> **Important**
>
> Cette procédure s'applique à l'installation ou la réinstallation de l'Agent SSM sur une instance EC2 pour Windows Server. Si vous avez besoin d'installer l'agent sur un serveur sur site ou sur une machine virtuelle (VM) afin qu'il puisse être utilisé avec Systems Manager, consultez Installer Agent SSM pour un environnement hybride (Windows) (p. 62).

Pour installer manuellement la dernière version de Agent SSM sur les instances EC2 pourWindows Server

1. Connectez-vous à votre instance en utilisant le Bureau à distance ou Windows PowerShell.
2. Téléchargez la version la plus récente de l'Agent SSM sur votre instance. Vous pouvez effectuer le téléchargement à l'aide PowerShell de commandes ou d'un lien de téléchargement direct.

   > **Note**
   >
   > Les URLs dans cette étape vous permettent Agent SSM de télécharger à partir de n'importe quelle AWS région. Si vous souhaitez télécharger l'agent à partir d'une région spécifique, utilisez plutôt une URL spécifique à la région :
   > `https://amazon-ssm-`*`region`*`.s3.`*`region`*`.amazonaws.com/latest/windows_amd64/AmazonSSMAgentSetup.exe`
   > *`région`* représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de *`region`* prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

   PowerShell

   Exécutez les trois PowerShell commandes suivantes dans l'ordre. Ces commandes vous permettent de télécharger l'Agent SSM sans ajuster les paramètres de sécurité renforcée d'Internet Explorer (IE), puis d'installer l'agent et de supprimer le fichier d'installation.

   64-bit

   ```
   Invoke-WebRequest `
       https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/
   windows_amd64/AmazonSSMAgentSetup.exe `
       -OutFile $env:USERPROFILE\Desktop\SSMAgent_latest.exe
   ```

   32-bit

   ```
   Invoke-WebRequest `
       https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/windows_386/
   AmazonSSMAgentSetup.exe `
       -OutFile $env:USERPROFILE\Desktop\SSMAgent_latest.exe
   ```

```
Start-Process `
    -FilePath $env:USERPROFILE\Desktop\SSMAgent_latest.exe `
    -ArgumentList "/S"
```

```
rm -Force $env:USERPROFILE\Desktop\SSMAgent_latest.exe
```

Téléchargement direct

Téléchargez la dernière version de l'Agent SSM sur votre instance en utilisant le lien suivant. Si vous le souhaitez, mettez à jour cette URL avec une URL spécifique à une AWS région.

https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/windows_amd64/AmazonSSMAgentSetup.exe

Exécutez le fichier `AmazonSSMAgentSetup.exe` téléchargé pour installer Agent SSM.

3. Démarrez ou redémarrez Agent SSM en envoyant la commande suivante dans PowerShell :

```
Restart-Service AmazonSSMAgent
```

Important

Agent SSMnécessite Windows PowerShell 3.0 ou version ultérieure pour exécuter certains SSM documents sur Windows Server les instances (par exemple, le document AWS-ApplyPatchBaseline). Vérifiez que vos instances Windows Server exécutent Windows Management Framework 3.0 ou version ultérieure. Cette infrastructure inclut Windows PowerShell . Pour plus d'informations, consultez Windows Management Framework 3.0.

Configurer l'Agent SSM pour utiliser un proxy pour les instances Windows Server

Les informations contenues dans cette rubrique concernent les instances Windows Server créées au plus tôt en novembre 2016 qui n'utilisent pas l'option d'installation Nano.

Si votre instance est une instance Windows Server 2008-2012 R2 créée avant novembre 2016, EC2Config traite les demandes Systems Manager sur votre instance. Nous vous recommandons de mettre à niveau vos instances existantes pour utiliser la dernière version d' EC2Config . Le programme EC2Config d'installation le plus récent vous permet d'installer Agent SSM côte à côte avec EC2Config . Cette version côte à côte de Agent SSM est compatible avec vos instances créées à partir d'anciennes versions de Windows AMIs et vous permet d'utiliser SSM les fonctions publiées après novembre 2016. Pour plus d'informations sur l'installation de la dernière version du EC2Config service, consultez Installation de la dernière version d' EC2Config dans le Amazon EC2 Guide de l'utilisateur pour les instances Windows . Si vous ne mettez pas à niveau vers la dernière version d' EC2Config et que vous utilisez EC2Config pour traiter Systems Manager les demandes, vous devez configurer les paramètres proxy pour EC2Config . Pour plus d'informations sur la configuration EC2Config d' pour utiliser un proxy, consultez Configuration des paramètres proxy pour le EC2Config service .

Pour configurer l'Agent SSM afin d'utiliser un proxy

1. A l'aide du Bureau à distance ou PowerShell de Windows, connectez-vous à l'instance que vous souhaitez configurer pour utiliser un proxy.
2. Exécutez le bloc de commande suivant dans PowerShell . Remplacez *hostname* et *port* avec les informations sur votre proxy :

```
$serviceKey = "HKLM:\SYSTEM\CurrentControlSet\Services\AmazonSSMAgent"
$keyInfo = (Get-Item -Path $serviceKey).GetValue("Environment")
$proxyVariables = @("http_proxy=hostname:port", "no_proxy=169.254.169.254")

If($keyInfo -eq $null)
{
New-ItemProperty -Path $serviceKey -Name Environment -Value $proxyVariables -
PropertyType MultiString -Force
} else {
Set-ItemProperty -Path $serviceKey -Name Environment -Value $proxyVariables
}
Restart-Service AmazonSSMAgent
```

Après avoir exécuté la commande précédente, vous pouvez consulter les journaux Agent SSM pour confirmer que les paramètres proxy ont été appliqués. Les entrées dans les journaux ressemblent à ce qui suit. Pour de plus amples informations sur les journaux Agent SSM, veuillez consulter Affichage des Agent SSM journaux (p. 107).

```
2020-02-24 15:31:54 INFO Getting IE proxy configuration for current user: The operation
 completed successfully.
2020-02-24 15:31:54 INFO Getting WinHTTP proxy default configuration: The operation
 completed successfully.
2020-02-24 15:31:54 INFO Proxy environment variables:
2020-02-24 15:31:54 INFO http_proxy: hostname:port
2020-02-24 15:31:54 INFO https_proxy:
2020-02-24 15:31:54 INFO no_proxy: 169.254.169.254
2020-02-24 15:31:54 INFO Starting Agent: amazon-ssm-agent - v2.3.871.0
2020-02-24 15:31:54 INFO OS: windows, Arch: amd64
```

Pour réinitialiser la configuration du proxy de l'Agent SSM

1. A l'aide du Bureau à distance ou PowerShell de Windows, connectez-vous à l'instance à configurer.
2. Si vous vous êtes connecté à l'aide des services Bureau à distance, lancez PowerShell en tant qu'administrateur.
3. Exécutez le bloc de commande suivant dans PowerShell :

```
Remove-ItemProperty -Path HKLM:\SYSTEM\CurrentControlSet\Services\AmazonSSMAgent -Name
 Environment
Restart-Service AmazonSSMAgent
```

Priorité du paramètre proxy Agent SSM

Lors de la configuration des paramètres proxy pour l'Agent SSM sur les instances Windows Server, il est important de comprendre que ces paramètres sont évalués et appliqués à la configuration de l'agent lors du démarrage de l'Agent SSM. La façon dont vous configurez vos paramètres de proxy pour une instance Windows Server peut déterminer si d'autres paramètres peuvent remplacer vos paramètres souhaités. Les paramètres proxy de l'Agent SSM sont évalués dans l'ordre suivant.

1. Paramètres AmazonSSMAgent du registre (`HKLM:\SYSTEM\CurrentControlSet\Services\AmazonSSMAgent`)
2. Variables d'environnement système (http_proxy, https_proxy, no_proxy)
3. variables LocalSystem d'environnement de compte utilisateur (http_proxy, https_proxy, no_proxy)
4. Paramètres WinHTTP proxy
5. Paramètres d'Internet Explorer

Paramètres du proxy de l'Agent SSM et services Systems Manager

Si vous avez configuré l' Agent SSM pour utiliser un proxy et que vous utilisez AWS Systems Manager des services , tels que la fonctionnalité Exécuter la commande et Gestionnaire de correctifs , qui utilisent PowerShell ou le client Windows Update lors de leur exécution sur Windows Server des instances , vous devez configurer des paramètres proxy supplémentaires. Sinon, l'opération peut échouer car les paramètres de proxy utilisés par PowerShell et le client Windows Update ne sont pas hérités de la configuration du Agent SSM proxy.

Pour Exécuter la commande, vous devez configurer les paramètres de proxy `WinINet` sur vos instances Windows Server. Les PowerShell commandes suivantes renvoient les paramètres proxy actuels et appliquent vos paramètres proxy à `WinINet` .`WinINet`

```
[System.Net.WebRequest]::DefaultWebProxy

$proxyServer = "http://hostname:port"
$proxyBypass = "169.254.169.*"
$WebProxy = New-Object System.Net.WebProxy($proxyServer,$true,$proxyBypass)

[System.Net.WebRequest]::DefaultWebProxy = $WebProxy
```

Pour Gestionnaire de correctifs , vous devez configurer les paramètres de proxy à l'échelle du système afin que le client Windows Update puisse rechercher et télécharger les mises à jour. Nous vous recommandons d'utiliser Exécuter la commande pour exécuter les commandes suivantes, car elles s'exécutent sur le compte SYSTEM et les paramètres s'appliquent à l'ensemble du système. Les commandes netsh suivantes renvoient les paramètres de proxy actuels et appliquent vos paramètres proxy au système local.

```
netsh winhttp show proxy

netsh winhttp set proxy hostname:port
```

Pour plus d'informations sur l'utilisation de Fonctionnalité Exécuter la commande, consultez Exécuter des commandes avec Systems Manager Fonctionnalité Exécuter la commande (p. 946).

Vérification de Agent SSM l'état et démarrage de l'agent

Cette rubrique répertorie les commandes permettant de vérifier si Agent SSM s'exécute sur chaque système d'exploitation pris en charge. Elle fournit également les commandes pour démarrer l'agent s'il n'est pas en cours d'exécution.

Système d'exploitation	Commande pour vérifier Agent SSM le statut	Commande pour démarrerAgent SSM
Amazon Linux	`sudo status amazon-ssm-agent`	`sudo start amazon-ssm-agent`

Amazon Linux 2	`sudo systemctl status amazon-ssm-agent`	`sudo systemctl enable amazon-ssm-agent`
		
		`sudo systemctl start amazon-ssm-agent`

Système d'exploitation	Commande pour vérifier Agent SSM le statut	Commande pour démarrerAgent SSM
CentOS6.x	`sudo status amazon-ssm-agent`	`sudo start amazon-ssm-agent`

CentOS7.x et CentOS 8.x	`sudo systemctl status amazon-ssm-agent`	`sudo systemctl enable amazon-ssm-agent`
		
		`sudo systemctl start amazon-ssm-agent`

Debian Server8, 9 et 10	`sudo systemctl status amazon-ssm-agent`	`sudo systemctl enable amazon-ssm-agent`
		
		`sudo systemctl start amazon-ssm-agent`

Oracle Linux	`sudo systemctl status amazon-ssm-agent`	`sudo systemctl enable amazon-ssm-agent`
		
		`sudo systemctl start amazon-ssm-agent`

Red Hat Enterprise Linux(RHEL) 6.x	`sudo status amazon-ssm-agent`	`sudo start amazon-ssm-agent`

Red Hat Enterprise Linux(RHEL) 7.x et 8.x	`sudo systemctl status amazon-ssm-agent`	`sudo systemctl enable amazon-ssm-agent`
		
		`sudo systemctl start amazon-ssm-agent`

SUSE Linux Enterprise Server (SLES)	`sudo systemctl status amazon-ssm-agent`	`sudo systemctl enable amazon-ssm-agent`
		
		`sudo systemctl start amazon-ssm-agent`

Ubuntu Server14.04 (tous) et 16.04 (32 bits)	`sudo status amazon-ssm-agent`	`sudo start amazon-ssm-agent`

Ubuntu ServerInstances 16.04 64 bits (installation d'un package de base de données)	`sudo systemctl status amazon-ssm-agent`	`sudo systemctl enable amazon-ssm-agent`
		
		`sudo systemctl start amazon-ssm-agent`

Ubuntu Server16.04, 18.04 et 20.04 LTS & et 20.10 STR 64 bits (installation de packages Snap)	`sudo snap list amazon-ssm-agent`	`sudo snap start amazon-ssm-agent`
Windows Server	Exécuter dans PowerShell : `Get-Service AmazonSSMAgent`	Exécuter en mode PowerShell Administrateur : `Start-Service AmazonSSMAgent`

Contenu connexe

- Installation et configuration de l'Agent SSM sur les instances EC2 pour Linux (p. 75)
- Installation et configuration Agent SSM de sur les instances EC2 pourWindows Server (p. 100)
- Vérification du numéro de Agent SSM version (p. 106)

Vérification du numéro de Agent SSM version

Certaines fonctionnalités Systems Manager comportent des prérequis incluant qu'une version minimale de l'Agent SSM soit installée sur vos instances gérées. Vous pouvez connaître la version de l'Agent SSM actuellement installée sur vos instances gérées à l'aide de la console AWS Systems Manager ou en vous connectant à vos instances.

Les procédures suivantes décrivent comment connaître la version de l'Agent SSM actuellement installée sur vos instances gérées.

Pour vérifier le numéro de version d' Agent SSM installé sur une instance

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Instances gérées.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Instances gérées.
3. Notez la version de l'agent.

Pour obtenir la version de l'Agent SSM actuellement installée à partir du système d'exploitation

Windows

1. Connectez-vous à votre instance .
2. Exécutez la commande suivante PowerShell.

```
Get-WmiObject Win32_Product | Where-Object {$_.Name -eq 'Amazon SSM Agent'} | Select-
Object Name,Version
```

Linux

> **Note**
>
> Cette commande varie en fonction du gestionnaire de package de votre système d'exploitation.

1. Connectez-vous à votre instance .
2. Exécutez la commande suivante pour Amazon Linux et Amazon Linux 2.

```
yum info amazon-ssm-agent
```

Nous vous recommandons d'utiliser la dernière version de l'Agent SSM afin de bénéficier des fonctionnalités nouvelles ou mises à jour. Pour vous assurer que vos instances gérées exécutent toujours la version la plus récente de l'Agent SSM, vous pouvez automatiser le processus de mise à jour de

l'Agent SSM. Pour plus d'informations, consultez la section Automatisation des mises à jour versAgent SSM (p. 109).

Affichage des Agent SSM journaux

L'Agent SSM écrit des informations relatives aux exécutions, commandes, actions planifiées, erreurs et états d'intégrité dans les fichiers journaux de chaque instance. Vous pouvez afficher les fichiers journaux en vous connectant manuellement à une instance ou vous pouvez envoyer automatiquement les journaux à Amazon CloudWatch Logs. Pour de plus amples informations sur l'envoi de journaux à CloudWatch, veuillez consulter Surveillance des AWS Systems Manager (p. 1326).

Vous pouvez afficher les journaux Agent SSM sur les instances aux emplacements ci-dessous.

Linux

```
/var/log/amazon/ssm/amazon-ssm-agent.log

/var/log/amazon/ssm/errors.log

/var/log/amazon/ssm/audits/amazon-ssm-agent-audit-YYYY-MM-DD
```

Windows

```
%PROGRAMDATA%\Amazon\SSM\Logs\amazon-ssm-agent.log

%PROGRAMDATA%\Amazon\SSM\Logs\errors.log

%PROGRAMDATA%\Amazon\SSM\Logs\audits\amazon-ssm-agent-audit-YYYY-MM-DD
```

Pour les instances Linux, les fichiers Agent SSM `sterr` et `stdout` sont écrits dans le répertoire suivant : /var/lib/amazon/ssm.

Pour de plus amples informations sur l'activation de la journalisation de débogage de Agent SSM, veuillez consulter Activation Agent SSM de la journalisation de débogage (p. 107).

Pour plus d'informations sur `cihub/seelog` la configuration, consultez le Wiki Seelog sur GitHub . Pour obtenir des exemples de `cihub/seelog` configurations, consultez les exemples de référentiel cihub/seelog sur GitHub.

Activation Agent SSM de la journalisation de débogage

Utilisez la procédure pour activer la journalisation de débogage Agent SSM sur Windows Server et les instances Linux gérées.

Linux

Pour activer la journalisation de débogage Agent SSM sur les instances Linux

1. Utilisez Systems Manager Session Manager pour vous connecter à l'instance sur laquelle vous souhaitez activer la journalisation de débogage, ou connectez-vous à l'instance gérée. Pour plus d'informations, consultez la section Utilisation des Session Manager (p. 917).
2. Effectuez une copie du fichier seelog.xml.template. Remplacez le nom de la copie par seelog.xml.

   Sur la plupart des types d'instance Linux, le fichier est situé dans le répertoire `/etc/amazon/ssm/seelog.xml.template` .

Sur Ubuntu Server 20.10 STR & 20.04, 18.04 et 16.04 LTS, le fichier se trouve dans le répertoire `/snap/amazon-ssm-agent/current/` .

3. Modifiez le fichier `seelog.xml` pour modifier le comportement de journalisation par défaut. Modifiez la valeur de minlevel de info pour déboguer, comme illustré dans l'exemple suivant.

   ```
   <seelog type="adaptive" mininterval="2000000" maxinterval="100000000"
   critmsgcount="500" minlevel="debug">
   ```

4. Redémarrez Agent SSM à l'aide de la commande suivante.

   ```
   sudo service amazon-ssm-agent restart
   ```

Windows

Pour activer la journalisation de débogage Agent SSM sur les instances Windows Server

1. Utilisez Systems Manager Session Manager pour vous connecter à l'instance sur laquelle vous souhaitez activer la journalisation de débogage, ou connectez-vous à l'instance gérée. Pour plus d'informations, consultez la section Utilisation des Session Manager (p. 917).
2. Effectuez une copie du fichier seelog.xml.template. Remplacez le nom de la copie par seelog.xml. Ce fichier se trouve dans le répertoire suivant.

   ```
   %PROGRAMFILES%\Amazon\SSM\seelog.xml.template
   ```

3. Modifiez le fichier `seelog.xml` pour modifier le comportement de journalisation par défaut. Modifiez la valeur de minlevel de info pour déboguer, comme illustré dans l'exemple suivant.

   ```
   <seelog type="adaptive" mininterval="2000000" maxinterval="100000000"
   critmsgcount="500" minlevel="debug">
   ```

4. Recherchez l'entrée suivante :

   ```
   filename="{{LOCALAPPDATA}}\Amazon\SSM\Logs\amazon-ssm-agent.log"
   ```

   Modifiez cette entrée pour utiliser le chemin d'accès suivant :

   ```
   filename="C:\ProgramData\Amazon\SSM\Logs\amazon-ssm-agent.log"
   ```

5. Recherchez l'entrée suivante :

   ```
   filename="{{LOCALAPPDATA}}\Amazon\SSM\Logs\errors.log"
   ```

   Modifiez cette entrée pour utiliser le chemin d'accès suivant :

   ```
   filename="C:\ProgramData\Amazon\SSM\Logs\errors.log"
   ```

6. Redémarrez Agent SSM à l'aide de la PowerShell commande suivante en mode Administrateur :

   ```
   Restart-Service AmazonSSMAgent
   ```

Restriction de l'accès aux commandes de niveau racine viaAgent SSM

L'Agent SSM s'exécute sur les instances EC2 en utilisant les autorisations racine (Linux) ou les autorisations SYSTEM (Windows Server). Parce qu'elles représentent le plus haut niveau de privilèges

d'accès au système, toute entité approuvée qui a reçu l'autorisation d'envoyer des commandes à l'Agent SSM possède des autorisations racine ou SYSTEM. (Dans AWS , une entité approuvée qui peut effectuer des actions et accéder à des ressources dans AWS est appelée mandataire. Un mandataire peut être un utilisateur racine d'un AWS compte , un IAM utilisateur ou un rôle .)

Ce niveau d'accès est nécessaire à un mandataire pour qu'il puisse envoyer des commandes Systems Manager autorisées pour l'Agent SSM, mais il permet également à un mandataire d'exécuter des codes malveillants en exploitant d'éventuelles vulnérabilités dans l'Agent SSM.

En particulier, les autorisations nécessaires pour exécuter les commandes SendCommand et StartSession doivent être soigneusement restreintes. Une première étape judicieuse consiste à accorder les autorisations à chaque commande uniquement pour sélectionner les mandataires dans votre organisation. Toutefois, nous vous recommandons de renforcer encore votre sécurité en limitant les instances sur lesquelles un mandataire peut exécuter ses commandes. Cela est possible dans la stratégie utilisateur IAM affectée au mandataire. Dans la stratégie IAM, vous pouvez inclure une condition qui limite l'utilisateur à l'exécution de commandes uniquement sur les instances balisées à l'aide de balises Amazon EC2 spécifiques ou de combinaisons de balises EC2.

Par exemple, supposons que vous avez deux parcs d'instances, l'un à des fins de test, l'autre à des fins de production. Dans la IAM stratégie appliquée aux jeunes ingénieurs, vous spécifiez qu'ils peuvent exécuter des commandes uniquement sur les instances balisées avec `ssm:resourceTag/testServer` . Mais pour un plus petit groupe d'ingénieurs principaux, qui doivent avoir accès à toutes les instances, vous accordez l'accès aux instances balisées avec `ssm:resourceTag/testServer` et `ssm:resourceTag/productionServer` .

Avec cette approche, si de jeunes ingénieurs essaient d'exécuter une commande sur une instance de production, l'accès leur est refusé car la stratégie IAM qui leur est affectée ne fournit pas un accès explicite aux instances balisées avec `ssm:resourceTag/productionServer`.

Pour de plus amples informations et d'exemples, veuillez consulter les rubriques suivantes :

- Restriction de l'accès de la fonctionnalité Fonctionnalité Exécuter la commande en fonction des balises d'instance (p. 943)
- Limiter l'accès à la session en fonction des balises d'instance (p. 892)

Automatisation des mises à jour versAgent SSM

AWS publie une nouvelle version de l'Agent SSM lorsque nous ajoutons ou mettons à jour des fonctionnalités Systems Manager. Si vos instances utilisent une ancienne version de l'agent, vous ne pouvez pas utiliser les nouvelles fonctionnalités, ni bénéficier des fonctionnalités mises à jour. Pour ces raisons, nous vous recommandons d'automatiser le processus de mise à jour de l'Agent SSM sur vos instances à l'aide de l'une des méthodes suivantes.

Method	Details (Détails)
Mise à jour automatisée en un clic sur toutes les instances (recommandé)	Vous pouvez configurer toutes les instances de votre AWS compte pour qu'elles recherchent et téléchargent automatiquement les nouvelles versions de Agent SSM . Pour ce faire, choisissez Agent auto update (Mise à jour automatique de l'agent) dans la page Managed instances (Instances gérées) de la console AWS Systems Manager, comme décrit plus loin dans cette rubrique.

Method	Details (Détails)
Mise à jour globale ou sélective	Vous pouvez utiliser State Manager pour créer une association qui télécharge et installe automatiquement l'Agent SSM sur vos instances. Si vous souhaitez limiter les interruptions de vos charges de travail, vous pouvez créer une fenêtre de maintenance Systems Manager pour effectuer l'installation pendant les périodes désignées. Les deux méthodes vous permettent de créer une configuration de mise à jour globale pour toutes vos instances ou de sélectionner les instances qui seront mises à jour. Pour de plus amples informations sur la création d'une association State Manager, veuillez consulter Mise à jour automatique d'Agent SSM (interface de ligne de commande) (p. 1043). Pour de plus amples informations sur l'utilisation d'une fenêtre de maintenance, veuillez consulter Mise à jour automatique de l'Agent SSM (AWS CLI) et Mise à jour automatique de l'Agent SSM (console).
Mise à jour globale ou sélective pour les nouveaux environnements	SI vous débutez avec Systems Manager, nous vous conseillons d'utiliser l'option Update Systems Manager (SSM) Agent every two weeks (Mettre à jour l'Agent Systems Manager (SSM) toutes les deux semaines) dans Systems Manager Quick Setup. Le programme d'installation rapide vous permet de créer une configuration de mise à jour globale pour toutes vos instances ou de sélectionner les instances qui seront mises à jour. Pour plus d'informations, consultez la section AWS Systems Manager Configuration rapide (p. 17).

Si vous préférez mettre à jour l'Agent SSM sur vos instances manuellement, vous pouvez vous abonner à des notifications que AWS publie lorsqu'une nouvelle version de l'agent est lancée. Pour plus d'informations, consultez la section Abonnement aux notifications Agent SSM (p. 111). Après vous être abonné aux notifications, vous pouvez utiliser l'option Run command (Exécuter la commande) pour mettre à jour manuellement une ou plusieurs instances à la dernière version. Pour plus d'informations, consultez la section Mettre à jour l'Agent SSM à l'aide de Fonctionnalité Exécuter la commande (p. 948).

Mise à jour automatiqueAgent SSM

Vous pouvez configurer Systems Manager pour mettre à jour automatiquement toutes les instances gérées de votre Agent SSM compte.AWS Si vous activez cette option, Systems Manager recherche automatiquement toutes les deux semaines une nouvelle version de l'agent. S'il existe une nouvelle version, met Systems Manager automatiquement à jour l'agent vers la dernière version publiée à l'aide du document SSM`AWS-UpdateSSMAgent`. Nous vous encourageons à choisir cette option pour vous assurer que vos instances exécutent toujours la version la plus récente de Agent SSM .

> **Note**
>
> Si vous utilisez une commande `yum` pour mettre à jour Agent SSM sur une instance gérée après l'installation ou la mise à niveau de l'agent à l'aide d'un document SSM`AWS-UpdateSSMAgent`, le message suivant pourrait s'afficher : « Avertissement : RPMDB a été modifiée hors de yum ». Ce message est attendu et peut être ignoré en toute sécurité.

Pour mettre à jour automatiquement l'Agent SSM

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Managed instances (Instances gérées).
3. Choisissez Agent auto update (Mise à jour automatique de l'agent).

Pour arrêter automatiquement le déploiement des versions mises à jour de l'Agent SSM sur toutes les instances gérées de votre compte, choisissez Delete (Supprimer) dans la page Managed instances (Instances gérées). Cette action supprime l'association State Manager qui met automatiquement à jour l'Agent SSM sur vos instances.

Abonnement aux notifications Agent SSM

Amazon Simple Notification Service (Amazon SNS) peut vous avertir en cas de publication de nouvelles versions de l'Agent SSM. Pour vous abonner à ces notifications, utilisez la procédure suivante.

> **Tip**
>
> Vous pouvez également vous abonner aux notifications en regardant la page Agent SSM Notes de mise à jour sur GitHub .

Pour s'abonner aux notifications Agent SSM

1. Ouvrez la console Amazon SNS à l'adresse https://console.aws.amazon.com/sns/v3/home.
2. Dans le sélecteur de région dans la barre de navigation, choisissez USA Est (Virginie du Nord), si elle n'est pas déjà sélectionnée. Vous devez sélectionner cette région, car les notifications SNS de l'Agent SSM auxquelles vous vous abonnez sont générées à partir de cette région uniquement.
3. Dans le panneau de navigation, choisissez Abonnements.
4. Choisissez Create subscription.
5. Pour Créer un abonnement, procédez comme suit :
   a. Pour ARN de la rubrique, utilisez l'Amazon Resource Name (ARN) suivant :

      `arn:aws:sns:us-east-1:720620558202:SSM-Agent-Update`
   b. Pour Protocole, choisissez `Email` ou `SMS`.
   c. Pour Point de terminaison, tapez une adresse e-mail que vous pouvez utiliser pour recevoir les notifications. Si vous choisissez `SMS`, saisissez un indicatif et un numéro.
   d. Choisissez Create subscription.
6. Si vous choisissez `Email`, vous recevrez un e-mail vous demandant de confirmer votre abonnement. Ouvrez l'e-mail et suivez les instructions pour terminer votre abonnement.

Chaque fois qu'une nouvelle version de l'Agent SSM est publiée, nous envoyons des notifications aux abonnés de la rubrique. Si vous ne souhaitez plus recevoir ces notifications, exécutez la procédure suivante pour annuler votre abonnement.

Pour annuler votre abonnement aux notifications Agent SSM

1. Ouvrez la console Amazon SNS.
2. Dans le panneau de navigation, choisissez Abonnements.
3. Sélectionnez l'abonnement, puis sous Actions, choisissez Supprimer des abonnements. Lorsque vous êtes invité à confirmer l'opération, choisissez Supprimer.

À propos des autorisations minimales relatives aux compartiments S3 pour l'Agent SSM

Cette rubrique contient des informations sur les compartiments Amazon Simple Storage Service (Amazon S3) auxquels l'Agent SSM peut avoir besoin d'accéder pour effectuer des opérations Systems Manager. Ces compartiments sont accessibles publiquement, mais dans certains cas, il se peut que vous ayez besoin de fournir une autorisation explicite dans un profil d'instance EC2 pour Systems Manager, ou dans un rôle de service pour les instances d'un environnement hybride. Le plus souvent, vous devez accorder ces autorisations si vous utilisez un point de terminaison de VPC privé dans vos opérations Systems Manager. Sinon, vos ressources ne peuvent pas accéder à ces compartiments publics.

Pour accorder l'accès à ces compartiments, vous devez créer une stratégie d'autorisations S3 personnalisée, puis l'attacher à votre profil d'instance (pour les instances EC2) ou à votre rôle de service (pour des serveurs sur site et les machines virtuelles dans un environnement hybride).

Pour les mises à jour de l'Agent SSM, si le profil d'instance ne fournit pas d'autorisations pour ces compartiments, Agent SSM effectue un appel HTTP pour télécharger la mise à jour.

> **Note**
>
> Ces autorisations ne permettent d'accéder aux compartiments gérés AWS requis par l'Agent SSM. Elles ne fournissent pas les autorisations qui sont nécessaires pour les autres opérations Amazon S3. Elles ne fournissent pas non plus l'autorisation sur vos propres compartiments S3.

Pour de plus amples informations, consultez les rubriques suivantes :

- Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34)
- Créer un rôle de service IAM pour un environnement hybride (p. 50)

Table des matières

- Autorisations requises (p. 112)
- Example (p. 114)

Autorisations requises

Le tableau suivant décrit chacune des autorisations de la stratégie Amazon S3 qui sont nécessaires pour utiliser Systems Manager.

Autorisations Amazon S3 requises par l'Agent SSM

ARN du compartiment S3	Description
`arn:aws:s3:::aws-ssm-`*`region`*`/*`	Fournit l'accès au compartiment S3 contenant les modules requis à utiliser avec les documents SSM. **Note** Dans la région Région Moyen-Orient (Bahrain) (me-south-1) uniquement, ce compartiment utilise une convention de dénomination différente. Pour cette AWS région uniquement, utilisez plutôt le compartiment suivant.

ARN du compartiment S3	Description
	• `aws-patch-manager-me-south-1-a53fc9dce`
`arn:aws:s3:::aws-windows-downloads-region/*`	Requise pour certains documents SSM qui prennent en charge les systèmes d'exploitation Windows.
`arn:aws:s3:::amazon-ssm-region/*`	Requise pour la mise à jour des installations de l'Agent SSM. Ces compartiments contiennent les packages d'installation de l'Agent SSM, et l'installation des manifestes qui sont référencés par le document et le plug-in `AWS-UpdateSSMAgent`. Si ces autorisations ne sont pas fournies, l'Agent SSM effectue un appel HTTP pour télécharger la mise à jour.
`arn:aws:s3:::amazon-ssm-packages-region/*`	Requise pour utiliser les versions de l'Agent SSM antérieures à la version 2.2.45.0 en vue d'exécuter le document `AWS-ConfigureAWSPackage`.
`arn:aws:s3:::region-birdwatcher-prod/*`	Permet d'accéder au service de distribution utilisé par la version 2.2.45.0 et les versions ultérieures de l'Agent SSM. Ce service est utilisé pour exécuter le document `AWS-ConfigureAWSPackage`. Cette autorisation est nécessaire pour toutes les AWS régions, sauf (af-south-1) et Région Afrique (Le Cap) (eu-south-1).Région Europe (Milan)
`arn:aws:s3:::aws-ssm-distributor-file-region/*`	Permet d'accéder au service de distribution utilisé par la version 2.2.45.0 et les versions ultérieures de l'Agent SSM. Ce service est utilisé pour exécuter le document `AWS-ConfigureAWSPackage`. Cette autorisation est nécessaire uniquement pour l' Région Afrique (Le Cap) (af-south-1) et l' Région Europe (Milan) (eu-south-1).
`arn:aws:s3:::aws-ssm-document-attachments-region/*`	Permet d'accéder au compartiment S3 contenant les packages Distributor appartenant à Amazon.

ARN du compartiment S3	Description
`arn:aws:s3:::patch-baseline-snapshot-`*`region`*`/*`	Fournit l'accès au compartiment S3 contenant les instantanés de référence de correctifs. Cette autorisation est requise si vous utilisez le document SSM `AWS-RunPatchBaseline` ou le document SSM hérité `AWS-ApplyPatchBaseline`. **Note** Dans la région Région Moyen-Orient (Bahrain) (me-south-1) uniquement, ce compartiment utilise une convention de dénomination différente. Pour cette AWS région uniquement, utilisez plutôt le compartiment suivant. • `patch-baseline-snapshot-me-south-1-uduvl7q8`

`région` représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de *`region`* prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

Example

L'exemple suivant illustre comment fournir l'accès aux compartiments S3 requis pour les opérations Systems Manager dans la région Région USA Est (Ohio) (us-east-2).

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": "*",
            "Action": "s3:GetObject",
            "Resource": [
                "arn:aws:s3:::aws-ssm-us-east-2/*",
                "arn:aws:s3:::aws-windows-downloads-us-east-2/*",
                "arn:aws:s3:::amazon-ssm-us-east-2/*",
                "arn:aws:s3:::amazon-ssm-packages-us-east-2/*",
                "arn:aws:s3:::us-east-2-birdwatcher-prod/*",
                "arn:aws:s3:::patch-baseline-snapshot-us-east-2/*"
            ]
        }
    ]
}
```

Important

Dans cette stratégie, nous vous recommandons d'éviter d'utiliser des caractères génériques (*) à la place des régions spécifiques. Par exemple, utilisez `arn:aws:s3:::aws-ssm-us-east-2/*` et n'utilisez pas `arn:aws:s3:::aws-ssm-*/*` . L'utilisation de caractères génériques pourrait fournir un accès aux compartiments S3 auxquels vous n'avez pas l'intention d'accorder l'accès. Si vous souhaitez utiliser le profil d'instance pour plusieurs régions, nous vous recommandons de répéter le premier bloc `Statement` pour chaque région.

Dépannage de Agent SSM

Si vous rencontrez des problèmes lors de l'exécution d'opérations sur vos instances gérées, il peut y avoir un problème avec Agent SSM . Utilisez les informations suivantes pour vous permettre de visualiser les fichiers journaux de l'Agent SSM et dépanner l'agent.

Rubriques

- nAgent SSM'est pas à jour (p. 115)
- Afficher les fichiers journaux Agent SSM (p. 115)

nAgent SSM'est pas à jour

Une version mise à jour de l'Agent SSM est publiée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Si une version antérieure de l'agent est en cours d'exécution sur une instance, certains processus de l'Agent SSM peuvent échouer. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir l'Agent SSM à jour sur vos instances. Pour de plus amples informations, veuillez consulter Automatisation des mises à jour versAgent SSM (p. 109). Pour être alerté des mises à jour de Agent SSM, abonnez-vous à la page Agent SSM Release Notes sur GitHub.

Afficher les fichiers journaux Agent SSM

Agent SSM consigne des informations dans les fichiers suivants. Les informations contenues dans ces fichiers peuvent également vous aider à résoudre les problèmes.

> **Note**
>
> Si vous choisissez d'afficher ces journaux à l'aide de l'Explorateur de fichiers Windows, n'oubliez pas d'activer l'affichage des fichiers masqués et fichiers système dans les options de dossier.

Sous Windows

- `%PROGRAMDATA%\Amazon\SSM\Logs\amazon-ssm-agent.log`
- `%PROGRAMDATA%\Amazon\SSM\Logs\errors.log`

Sous Linux

- `/var/log/amazon/ssm/amazon-ssm-agent.log`
- `/var/log/amazon/ssm/errors.log`

Intégrations de produits et services avec Systems Manager

Par défaut, Systems Manager s'intègre à un certain nombre de AWS services et à d'autres produits et services. Les informations suivantes peuvent vous aider à configurer Systems Manager pour l'intégrer aux produits et services que vous utilisez.

- Intégration aux AWS services (p. 116)
- Intégration à d'autres produits et services (p. 138)
- Exemples d'intégration issus de la communauté (p. 149)

Intégration aux AWS services

Grâce à l'utilisation de `Command` et de `Automation` documents de SSM type, vous pouvez utiliser Systems Manager pour intégrer à de nombreux services AWS. Pour plus d'informations sur ces SSM documents, consultez Documents AWS Systems Manager (p. 1211) .

Systems Managerest intégré aux AWS services suivants.

Compute

Amazon Elastic Compute Cloud (Amazon EC2)	Amazon EC2fournit une capacité de calcul évolutive dans le cloud Amazon Web Services (AWS). L'utilisation d'Amazon EC2 vous dispense d'investir à l'avance dans du matériel et, par conséquent, vous pouvez développer et déployer les applications plus rapidement. Vous pouvez utiliser Amazon EC2 pour lancer autant de serveurs virtuels que nécessaire, configurer la sécurité et la mise en réseau, et gérer le stockage. vous Systems Manager permet d'effectuer plusieurs tâches sur Amazon EC2 les instances . Par exemple, vous pouvez lancer, configurer, gérer, gérer, dépanner et vous connecter en toute sécurité à vos Amazon EC2 instances. Vous pouvez également utiliser Systems Manager pour déployer des logiciels, évaluer la conformité et collecter l'inventaire à partir de vos Amazon EC2 instances. En savoir plus • Instances gérées AWS Systems Manager (p. 851) • AWS Systems Manager State Manager (p. 987) • AWS Systems Manager Fonctionnalité Exécuter la commande (p. 941)

	• AWS Systems Manager Gestionnaire de correctifs (p. 1047) • AWS Systems Manager Session Manager (p. 869) • AWS Systems Manager Distributor (p. 1174) • Conformité de la configuration AWS Systems Manager (p. 778) • Inventaire AWS Systems Manager (p. 791)
Amazon EC2 Auto Scaling	vous Auto Scaling permet de vous assurer que vous disposez du bon nombre d'instances Amazon EC2 disponibles pour gérer la charge de l'application. Vous créez des ensembles d'instances EC2, appelés groupes Auto Scaling. vous Systems Manager permet d'automatiser les procédures courantes, telles que l'application de correctifs à l'Amazon Machine Image (AMI) utilisée dans votre Auto Scaling modèle pour votre Auto Scaling groupe. En savoir plus Procédure : Appliquer des correctifs à une AMI et mettre à jour un groupe Auto Scaling (p. 651)
Amazon Elastic Container Service (Amazon ECS)	Amazon ECS est un service de gestion de conteneurs hautement évolutif et rapide, qui permet d'exécuter, d'arrêter et de gérer facilement des conteneurs Docker sur un cluster. vous Systems Manager permet de gérer des instances de conteneur à distance et d'injecter des données sensibles dans vos conteneurs en stockant vos données sensibles dans AWS Systems Manager Parameter Store des paramètres, puis en les référençant dans votre définition de conteneur. En savoir plus • Gestion des instances de conteneur à distance à l'aide d' AWS Systems Manager • Spécification de données sensibles à l'aide de Systems Manager Parameter Store

AWS Lambda	Lambdaest un service de calcul qui vous permet d'exécuter du code sans mettre en service ou gérer des serveurs. AWS Lambda exécute votre code uniquement lorsque cela est nécessaire et s'adapte automatiquement, qu'il s'agisse de quelques demandes par jour ou de milliers de demandes par seconde. vous Systems Manager permet d'utiliser Lambda des fonctions dans le contenu d'un document Automation à l'aide de l'`aws:invokeLambdaFunction`action. En savoir plus Procédure : Simplification de l'application de correctifs à l'AMI avec Automation, AWS Lambda et Parameter Store (p. 645)

Storage

Amazon Simple Storage Service (Amazon S3)	Amazon S3 est un service de stockage sur Internet. Destiné aux développeurs, il est conçu pour faciliter les opérations de calcul à l'échelle du web. Amazon S3 dispose d'une interface de services web simple que vous pouvez utiliser pour stocker et récupérer n'importe quel volume de données, à tout moment et n'importe où sur le web. vous Systems Manager permet d'exécuter des scripts et des Systems Manager documents distants qui sont stockés dans Amazon S3 . Distributor utilise Amazon S3 pour stocker les packages. Vous pouvez également envoyer la sortie pour Fonctionnalité Exécuter la commande et Session Manager vers Amazon S3 . En savoir plus • Exécution de scripts depuis Amazon S3 (p. 126) • Exécution de documents Systems Manager de commande à partir d'emplacements distants (p. 1286) • AWS Systems Manager Distributor (p. 1174) • Journalisation des données de session avec Amazon S3 (console) (p. 931)

Outils pour développeurs

AWS CodeBuild	CodeBuild est un service de génération entièrement géré dans le cloud. CodeBuild compile

	votre code source, exécute des tests unitaires et produit des artefacts prêts à être déployés. CodeBuild supprime la nécessité de mettre en service, de gérer et de dimensionner vos serveurs de génération. vous Parameter Store permet de stocker des informations sensibles pour vos spécifications de génération et vos projets. En savoir plus • Référence des spécifications de génération pour CodeBuild • Création d'un projet de génération dans AWS CodeBuild

Sécurité, identité et conformité

AWS Identity and Access Management (IAM)	IAMest un service web qui vous permet de contrôler en toute sécurité l'accès aux ressources AWS. Vous pouvez utiliser IAM pour contrôler les personnes qui s'authentifient (connectées) et sont autorisées (disposent d'autorisation) à utiliser les ressources. vous Systems Manager permet de contrôler l'accès aux services à l'aide d' IAM . En savoir plus • Fonctionnement d'AWS Systems Manager avec IAM (p. 1298) • Actions, ressources et clés de condition pour AWS Systems Manager • Créer des utilisateurs et des groupes IAM non administrateurs pour Systems Manager (p. 30) • Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34)
AWS Secrets Manager	Secrets Managerfacilite la gestion des secrets. Les secrets peuvent être des informations d'identification de base de données, des mots de passe, des clés d'API tierces et même un texte arbitraire. vous Parameter Store permet de récupérer des Secrets Manager secrets lors de l'utilisation d'autres services AWS qui prennent déjà en charge les références aux Parameter Store paramètres. En savoir plus

	Référencement des secrets AWS Secrets Manager à partir des paramètres Parameter Store (p. 134)
AWS Security Hub	vous Security Hub offre une vue complète de vos alertes de sécurité hautement prioritaires et de l'état de conformité de l'ensemble des AWS comptes. Security Hub regroupe, organise et hiérarchise vos alertes de sécurité, ou conclusions, à partir de plusieurs AWS services. Lorsque vous activez l'intégration entre Security Hub et Gestionnaire de correctifs , Security Hub surveille l'état d'application des correctifs de vos flottes du point de vue de la sécurité. Les détails de conformité des correctifs sont automatiquement exportés vers Security Hub . Cela vous permet d'utiliser une vue unique pour surveiller de manière centralisée l'état de conformité de vos correctifs et suivre d'autres résultats de sécurité. Vous pouvez recevoir des alertes lorsque des instances de votre parc sortent de la conformité des correctifs et vérifier les résultats de conformité des correctifs dans la Security Hub console . Des frais sont facturés pour l'utilisation d' Security Hub . Pour de plus amples informations, veuillez consulter Tarification de Security Hub. En savoir plus • Intégration d'Gestionnaire de correctifs à AWS Security Hub (p. 1145) • Présentation AWS Security Hub de l'AWS Security Hub Guide de l'utilisateur

Chiffrement et ICP

AWS Key Management Service (AWS KMS)	AWS KMSest un service géré qui facilite la création et le contrôle des clés principales client (clés CMK), les clés de chiffrement utilisées pour chiffrer vos données. vous Systems Manager permet d'utiliser AWS KMS pour créer `SecureString` des paramètres et chiffrer des données Session Manager de session. En savoir plus • Comment AWS Systems Manager Parameter Store utilise AWS KMS • Activer le chiffrement de clé AWS KMS des données de session (console) (p. 902)

Gestion et gouvernance

AWS CloudFormation	AWS CloudFormationest un service qui vous permet de modéliser et de configurer vos ressources Amazon Web Services afin de passer moins de temps à gérer ces ressources et de consacrer plus de temps à vos applications exécutées dans AWS. Parameter Storeest une source pour les références dynamiques. Les références dynamiques vous offrent un moyen compact et puissant de spécifier des valeurs externes qui sont stockées et gérées dans d'autres services de vos modèles de AWS CloudFormation pile. En savoir plus Utilisation de références dynamiques pour spécifier les valeurs de modèle
AWS CloudTrail	CloudTrailest un service AWS qui vous permet d'assurer la gouvernance, la conformité, ainsi que l'audit opérationnel et des risques de votre compte AWS. Les actions effectuées par un utilisateur, un rôle ou un service AWS sont enregistrées en tant qu'événements dans CloudTrail . Les événements incluent les actions réalisées dans AWS Management Console , AWS Command Line Interface et AWS SDKs et APIs . Systems Managers'intègre à CloudTrail , ce qui vous permet de capturer tous les appels d'API pour Systems Manager en tant qu'événements, y compris les appels depuis la Systems Manager console et les appels de code vers l' Systems Manager APIs . En savoir plus Journalisation des appels d'API AWS Systems Manager avec AWS CloudTrail (p. 1336)
Amazon CloudWatch Logs	vous Amazon CloudWatch Logs permet de centraliser les journaux de tous vos systèmes, applications et services AWS que vous utilisez. Vous pouvez ensuite facilement les afficher, les rechercher des codes ou des modèles d'erreur spécifiques, les filtrer en fonction de champs spécifiques ou les archiver en toute sécurité pour des analyses ultérieures. Systems Managerprend en charge l'envoi de journaux pour , et Agent SSM vers Fonctionnalité Exécuter la commande .Session ManagerCloudWatch Logs

	En savoir plus • Envoi de journaux d'instance à CloudWatch Logs (Agent CloudWatch) (p. 1327) • Configuration d'Amazon CloudWatch Logs pour Fonctionnalité Exécuter la commande (p. 1338) • Journalisation des données de session avec Amazon CloudWatch Logs (console) (p. 932)
Amazon EventBridge	EventBridgefournit un flux d'événements système en temps quasi réel qui décrivent les modifications apportées aux ressources Amazon Web Services (AWS). A l'aide de règles simples que vous pouvez configurer rapidement, vous pouvez faire correspondre des événements et les acheminer vers un ou plusieurs flux ou fonctions cibles. EventBridge prend connaissance des modifications opérationnelles au fur et à mesure qu'elles se produisent. EventBridge répond à ces modifications opérationnelles et prend des mesures correctives si nécessaire. Ces actions incluent l'envoi de messages pour répondre à l'environnement, l'activation de fonctions et la capture d'informations d'état. Systems Managerpossède plusieurs événements qui sont pris en charge en vous EventBridge permettant d'effectuer des actions en fonction du contenu de ces événements. En savoir plus Surveillance des événements Systems Manager avec Amazon EventBridge (p. 1340) Note Amazon EventBridge est le meilleur moyen de gérer vos événements. CloudWatch Events et EventBridge sont les mêmes services sous-jacents et API, mais EventBridge fournit plus de fonctions. Les modifications que vous apportez à CloudWatch ou à EventBridge apparaîtront dans chaque console. Pour de plus amples informations, veuillez consulter Guide de l'utilisateur Amazon EventBridge.

AWS Config	AWS Config fournit une vue détaillée de la configuration des ressources AWS dans votre compte AWS. Cela inclut la façon dont les ressources sont liées entre elles et la façon dont elles ont été configurées. Cela vous permet de voir comment les configurations et les relations changent au fil du temps. Systems Managerest intégré à AWS Config , ce qui fournit plusieurs règles vous permettant d'obtenir une meilleure visibilité sur vos Amazon EC2 instances. Ces règles vous aident à identifier les Amazon EC2 instances gérées par Systems Manager , les configurations du système d'exploitation, les mises à jour au niveau du système, les applications installées, les configurations réseau, etc. En savoir plus • AWS Config Types de ressources et relations entre ressources pris en charge par • Enregistrement de la configuration logicielle pour les instances gérées • Affichage de l'historique d'inventaire et suivi des modifications (p. 836)
AWS Trusted Advisor	Trusted Advisorest un outil en ligne qui fournit des conseils en temps réel pour vous aider à mettre en service vos ressources en suivant les bonnes pratiques AWS. Les Systems Manager hôtes Trusted Advisor et vous pouvez afficher les Trusted Advisor données dans Explorer . En savoir plus • Tableaux de bord Personal Health et Trusted Advisor hébergés par Systems Manager (p. 210) • AWS Systems Manager Explorer (p. 153)

Mise en réseau et diffusion de contenu

AWS PrivateLink	vous AWS PrivateLink permet de connecter de manière privée votre VPC aux services AWS pris en charge et aux services de point de terminaison de VPC sans avoir besoin d'une passerelle Internet, d'un périphérique NAT, d'une connexion VPN ou dAWS Direct Connect'une connexion. Systems Managerprend en charge les instances gérées qui se connectent à à Systems Manager

	APIs l'aide d' AWS PrivateLink . Cela améliore la sécurité de vos instances gérées depuis AWS PrivateLink qu' restreint l'ensemble du trafic réseau entre vos instances gérées, et Systems Manager vers le réseau Amazon.Amazon EC2 Cela signifie que les instances gérées ne sont pas obligées d'avoir accès à Internet. En savoir plus (Facultatif) Créer un point de terminaison de VPC (p. 42)

Analytics

Amazon Athena	Athena est un service de requête interactif qui facilite l'analyse des données directement dans Amazon Simple Storage Service (Amazon S3) à l'aide de la syntaxe SQL standard. Quelques actions suffisent dans pour pointer AWS Management Console vers vos données stockées dans Athena et commencer à utiliser le SQL standard pour exécuter des requêtes ad hoc et obtenir des résultats en quelques secondes.Amazon S3 Systems ManagerL'inventaire s'intègre Athena à pour vous aider à interroger les données d'inventaire à partir de plusieurs régions et comptes AWS. Athena L'intégration utilise la synchronisation des données de ressource afin que vous puissiez afficher les données d'inventaire à partir de toutes vos instances gérées sur la page Detailed View (Vue détaillée) de la console Systems Manager d'inventaire. En savoir plus • Interrogation des données d'inventaire à partir de plusieurs régions et comptes (p. 811) • Procédure : Utiliser la synchronisation de données de ressources pour regrouper les données d'inventaire (p. 843)
AWS Glue	AWS Glueest un service ETL (extraction, transformation et chargement) entièrement géré qui permet de classer vos données par catégorie, de les nettoyer, de les enrichir et de les déplacer de manière fiable entre différents magasins de données et flux de données. Systems Managerutilise AWS Glue pour analyser les données d'inventaire dans votre compartiment Amazon Simple Storage Service ().Amazon S3

	En savoir plus Interrogation des données d'inventaire à partir de plusieurs régions et comptes (p. 811)
Amazon QuickSight	Amazon QuickSight est un service d'analyse décisionnelle que vous pouvez utiliser pour créer des visualisations, effectuer des analyses ad hoc et obtenir des perspectives professionnelles à partir de vos données. Il peut découvrir automatiquement les sources de données AWS et fonctionne également avec vos sources de données. La synchronisation des données de Systems Manager ressource envoie les données d'inventaire collectées à partir de toutes vos instances gérées vers un seul Amazon S3 compartiment . Vous pouvez utiliser Amazon QuickSight pour interroger et analyser les données agrégées. En savoir plus • Configuration de la synchronisation de données de ressource pour l'inventaire (p. 800) • Procédure : Utiliser la synchronisation de données de ressources pour regrouper les données d'inventaire (p. 843)

Intégration d'applications

Amazon Simple Notification Service (Amazon SNS)	Amazon SNSest un service web qui coordonne et gère la diffusion ou l'envoi de messages aux points de terminaison ou aux clients abonnés. Systems Managergénère des statuts pour plusieurs services qui peuvent être capturés par Amazon SNS des notifications. En savoir plus • Surveillance des changements de statut Systems Manager à l'aide de notifications Amazon SNS (p. 1344) • Configuration de notifications ou d'actions de déclenchement basées sur des événements Parameter Store (p. 239)

AWS Management Console

Groupes de ressources AWS	Resource Groupsorganisez vos ressources AWS. Les groupes de ressources facilitent la gestion, le contrôle et l'automatisation des

	tâches simultanément sur un grand nombre de ressources. Des types de Systems Manager ressource tels que les instances gérées, SSM les documents, les fenêtres de maintenance, Parameter Store les paramètres et les références de correctifs peuvent être ajoutés aux groupes de ressources. En savoir plus Resource Groups dans AWS Systems Manager (p. 212)

Rubriques

- Exécution de scripts depuis Amazon S3 (p. 126)
- Référencement des secrets AWS Secrets Manager à partir des paramètres Parameter Store (p. 134)

Exécution de scripts depuis Amazon S3

Cette section décrit comment télécharger et exécuter des scripts à partir de Amazon S3. Vous pouvez exécuter différents types de scripts, y compris des manuels stratégiques Ansible, Python, Ruby, Shell et PowerShell .

Vous pouvez également télécharger un répertoire contenant plusieurs scripts. Lorsque vous exécutez le script principal dans le répertoire, exécute Systems Manager également tous les scripts référencés qui sont inclus dans le répertoire.

Notez les informations importantes suivantes relatives à l'exécution de scripts à partir de Amazon S3.

- Systems Managerne vérifie pas que votre script est capable de s'exécuter sur une instance. Avant de télécharger et d'exécuter le script, vous devez vérifier que le logiciel requis est installé sur l'instance. Vous pouvez également créer un document composite qui installe le logiciel par l'intermédiaire de Fonctionnalité Exécuter la commande ou de State Manager, puis télécharge et exécute le script.
- Vérifiez que votre compte utilisateur AWS Identity and Access Management (IAM), votre rôle ou votre groupe dispose de l'autorisation de lecture sur le compartiment S3.
- Assurez-vous que le profil d'instance de vos instances EC2 dispose des autorisations `s3:ListBucket` et `s3:GetObject`. Si le profil d'instance ne dispose pas de ces autorisations, le système ne parvient pas à télécharger votre script à partir du compartiment S3. Pour de plus amples informations, veuillez consulter Utilisation de profils d'instance dans le IAM Guide de l'utilisateur.

Rubriques

- Exécuter les scripts Ruby depuis Amazon S3 (p. 126)
- Exécuter des scripts shell depuis Amazon S3 (p. 129)
- Exécuter un PowerShell script depuisAmazon S3 (p. 131)

Exécuter les scripts Ruby depuis Amazon S3

Cette section présente les procédures vous permettant d'exécuter des scripts Ruby à partir de Amazon S3 à l'aide de la console Systems Manager ou de l'AWS CLI.

Exécuter un script Ruby depuis Amazon S3 (console)

Exécuter un script Ruby depuis Amazon S3

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fonctionnalité Exécuter la commande.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Fonctionnalité Exécuter la commande.
3. Choisissez Exécuter une commande.
4. Dans la liste Document de commande, choisissez AWS-RunRemoteScript.
5. Dans Paramètres de la commande, procédez comme suit :
   - Dans Type de source, sélectionnez S3.
   - Dans la zone de texte Infos sur la source, entrez les informations requises pour accéder à la source en respectant le format suivant :

     ```
     {"path":"https://s3.amazonaws.com/path_to_script"}
     ```

     Exemple :

     ```
     {"path":"https://s3.amazonaws.com/doc-example-bucket/scripts/ruby/helloWorld.rb"}
     ```
   - Dans le champ Ligne de commande, entrez les paramètres d'exécution du script. Voici un exemple de .

     ```
     helloWorld.rb argument-1 argument-2
     ```
   - (Facultatif) Dans le champ Répertoire de travail, saisissez le nom d'un répertoire de l'instance où vous souhaitez télécharger et exécuter le script.
   - (Facultatif) Dans Execution Timeout (Délai d'exécution), précisez le nombre de secondes durant lesquelles le système doit attendre avant de mettre en échec l'exécution de la commande de script.
6. Dans la section Cibles, identifiez les instances sur lesquelles vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant manuellement les instances ou en indiquant un groupe de ressources.

   > **Note**
   >
   > Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.
7. Pour Autres paramètres :
   - Pour Comment (Commentaire), saisissez des informations à propos de cette commande.
   - Pour Délai (secondes), précisez le nombre de secondes durant lesquelles le système doit attendre avant de mettre en échec l'exécution de la commande globale.
8. (Facultatif) Pour Contrôle du débit :
   - Pour Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage d'instances sur lesquelles exécuter simultanément la commande.

Note

Si vous avez sélectionné des cibles en indiquant des balises appliquées à des instances gérées ou en indiquant des groupes de ressources AWS, et que vous n'êtes pas sûr du nombre d'instances ciblées, limitez le nombre d'instances pouvant exécuter simultanément le document en spécifiant un pourcentage.

- Pour Seuil d'erreur, spécifiez quand arrêter l'exécution de la commande sur d'autres instances après son échec sur un nombre ou un pourcentage d'instances. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les instances sur lesquelles la commande est en cours de traitement peuvent également envoyer des erreurs.

9. (Facultatif) Pour Output options (Options de sortie), pour enregistrer la sortie de la commande dans un fichier, cochez la case Write command output to an S3 bucket (Écrire la sortie de commande vers un compartiment S3). Entrez les noms de compartiment et de préfixe (dossier) dans les zones.

   Note

   Les autorisations S3 qui accordent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour de plus amples informations, veuillez consulter Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). En outre, si le compartiment S3 spécifié se trouve dans un compte AWS différent, assurez-vous que le profil d'instance associé à l'instance dispose des autorisations nécessaires pour écrire dans ce compartiment.

10. Dans la section SNS notifications (Notifications SNS), si vous souhaitez que des notifications sur le statut de l'exécution de la commande soient envoyées, cochez la case Enable SNS notifications (Activer les notifications SNS).

    Pour plus d'informations sur la configuration des notifications Amazon SNS pour la fonctionnalité Exécuter la commande, consultez Surveillance des changements de statut Systems Manager à l'aide de notifications Amazon SNS (p. 1344).

11. Cliquez sur Exécuter.

Exécution d'un script Ruby depuis Amazon S3 à l'aide de l'AWS CLI

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Exécutez la commande suivante .

   Linux

```
aws ssm send-command \
    --document-name "AWS-RunRemoteScript" \
    --targets "Key=instanceids,Values=instance-IDs" \
    --parameters '{"sourceType":["S3"],"sourceInfo":["{\"path\":\"https://
s3.amazonaws.com/path_to_script\"}"],"commandLine":["script_name_and_arguments"]}'
```

   Voici un exemple de .

```
aws ssm send-command \
    --document-name "AWS-RunRemoteScript" \
    --targets "Key=instanceids,Values=i-02573cafcfEXAMPLE" \
```

```
    --parameters '{"sourceType":["S3"],"sourceInfo":["{\"path\":\"https://
s3.amazonaws.com/doc-example-bucket/scripts/ruby/helloWorld.rb\"}"],"commandLine":
["helloWorld.rb argument-1 argument-2"]}'
```

Windows

```
aws ssm send-command ^
    --document-name "AWS-RunRemoteScript" ^
    --targets "Key=instanceids,Values=instance-IDs" ^
    --parameters "sourceType"="S3",sourceInfo='{\"path\":\"https://
s3.amazonaws.com/path_to_script\"}',"commandLine"="script_name_and_arguments"
```

Voici un exemple de .

```
aws ssm send-command ^
    --document-name "AWS-RunRemoteScript" ^
    --targets "Key=instanceids,Values=i-02573cafcfEXAMPLE" ^
    --parameters "sourceType"="S3",sourceInfo='{\"path\":\"https://
s3.amazonaws.com/doc-example-bucket/scripts/ruby/helloWorld.rb
\"}',"commandLine"="helloWorld.rb argument-1 argument-2"
```

Exécuter des scripts shell depuis Amazon S3

Cette section inclut les procédures vous permettant d'exécuter des scripts shell à partir d' à l'aide de la Amazon S3 console ou de l' Systems Manager .AWS CLI

Exécuter un script Shell depuis Amazon S3 (console)

Exécuter un script Shell depuis Amazon S3

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fonctionnalité Exécuter la commande.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Fonctionnalité Exécuter la commande.
3. Choisissez Exécuter une commande.
4. Dans la liste Document de commande, choisissez AWS-RunRemoteScript.
5. Dans Paramètres de la commande, procédez comme suit :

   - Dans Type de source, sélectionnez S3.
   - Dans la zone de texte Infos sur la source, entrez les informations requises pour accéder à la source en respectant le format suivant :

     ```
     {"path":"https://s3.amazonaws.com/path_to_script"}
     ```

     Exemple :

     ```
     {"path":"https://s3.amazonaws.com/doc-example-bucket/scripts/shell/helloWorld.sh"}
     ```
   - Dans le champ Ligne de commande, entrez les paramètres d'exécution du script. Voici un exemple de .

```
helloWorld.sh argument-1 argument-2
```

- (Facultatif) Dans le champ Répertoire de travail, saisissez le nom d'un répertoire de l'instance où vous souhaitez télécharger et exécuter le script.
- (Facultatif) Dans Execution Timeout (Délai d'exécution), précisez le nombre de secondes durant lesquelles le système doit attendre avant de mettre en échec l'exécution de la commande de script.

6. Dans la section Cibles, identifiez les instances sur lesquelles vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant manuellement les instances ou en indiquant un groupe de ressources.

   Note

   Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.

7. Pour Autres paramètres :

   - Pour Comment (Commentaire), saisissez des informations à propos de cette commande.
   - Pour Délai (secondes), précisez le nombre de secondes durant lesquelles le système doit attendre avant de mettre en échec l'exécution de la commande globale.

8. (Facultatif) Pour Contrôle du débit :

   - Pour Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage d'instances sur lesquelles exécuter simultanément la commande.

     Note

     Si vous avez sélectionné des cibles en indiquant des balises appliquées à des instances gérées ou en indiquant des groupes de ressources AWS, et que vous n'êtes pas sûr du nombre d'instances ciblées, limitez le nombre d'instances pouvant exécuter simultanément le document en spécifiant un pourcentage.

   - Pour Seuil d'erreur, spécifiez quand arrêter l'exécution de la commande sur d'autres instances après son échec sur un nombre ou un pourcentage d'instances. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les instances sur lesquelles la commande est en cours de traitement peuvent également envoyer des erreurs.

9. (Facultatif) Pour Output options (Options de sortie), pour enregistrer la sortie de la commande dans un fichier, cochez la case Write command output to an S3 bucket (Écrire la sortie de commande vers un compartiment S3). Entrez les noms de compartiment et de préfixe (dossier) dans les zones.

   Note

   Les autorisations S3 qui accordent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour de plus amples informations, veuillez consulter Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). En outre, si le compartiment S3 spécifié se trouve dans un compte AWS différent, assurez-vous que le profil d'instance associé à l'instance dispose des autorisations nécessaires pour écrire dans ce compartiment.

10. Dans la section SNS notifications (Notifications SNS), si vous souhaitez que des notifications sur le statut de l'exécution de la commande soient envoyées, cochez la case Enable SNS notifications (Activer les notifications SNS).

    Pour plus d'informations sur la configuration des notifications Amazon SNS pour la fonctionnalité Exécuter la commande, consultez Surveillance des changements de statut Systems Manager à l'aide de notifications Amazon SNS (p. 1344).

11. Cliquez sur Exécuter.

Exécuter un script Shell depuis Amazon S3 à l'aide de l'AWS CLI

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).
2. Exécutez la commande suivante .

   Linux

   ```
   aws ssm send-command \
       --document-name "AWS-RunRemoteScript" \
       --targets "Key=instanceids,Values=instance-IDs" \
       --parameters '{"sourceType":["S3"],"sourceInfo":["{\"path\":\"https://
   s3.amazonaws.com/path_to_script\"}"],"commandLine":["script_name_and_arguments"]}'
   ```

   Voici un exemple de .

   ```
   aws ssm send-command \
       --document-name "AWS-RunRemoteScript" \
       --targets "Key=instanceids,Values=i-02573cafcfEXAMPLE" \
       --parameters '{"sourceType":["S3"],"sourceInfo":["{\"path\":\"https://
   s3.amazonaws.com/doc-example-bucket/scripts/shell/helloWorld.sh\"}"],"commandLine":
   ["helloWorld.sh argument-1 argument-2"]}'
   ```

   Windows

   ```
   aws ssm send-command ^
       --document-name "AWS-RunRemoteScript" ^
       --targets "Key=instanceids,Values=instance-IDs" ^
       --parameters "sourceType"="S3",sourceInfo='{\"path\":\"https://
   s3.amazonaws.com/path_to_script\"}","commandLine"="script_name_and_arguments"
   ```

   Voici un exemple de .

   ```
   aws ssm send-command ^
       --document-name "AWS-RunRemoteScript" ^
       --targets "Key=instanceids,Values=i-02573cafcfEXAMPLE" ^
       --parameters "sourceType"="S3",sourceInfo='{\"path\":\"https://
   s3.amazonaws.com/doc-example-bucket/scripts/shell/helloWorld.sh
   \"}","commandLine"="helloWorld.sh argument-1 argument-2"
   ```

Exécuter un PowerShell script depuisAmazon S3

Cette section inclut les procédures vous permettant d'exécuter PowerShell des scripts à partir d' à l'aide de la Amazon S3 console ou de l' Amazon EC2 .AWS CLI

Exécution d'un PowerShell script à partir dAmazon S3' (console)

Exécuter un PowerShell script depuisAmazon S3

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fonctionnalité Exécuter la commande.

   -ou-

Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Fonctionnalité Exécuter la commande.

3. Choisissez Exécuter une commande.
4. Dans la liste Document de commande, choisissez AWS-RunRemoteScript.
5. Dans Paramètres de la commande, procédez comme suit :
   - Dans Type de source, sélectionnez S3.
   - Dans la zone de texte Infos sur la source, entrez les informations requises pour accéder à la source en respectant le format suivant :

     ```
     {"path": "https://s3.amazonaws.com/path_to_script"}
     ```

     Exemple :

     ```
     {"path": "https://s3.amazonaws.com/DOC-EXAMPLE-BUCKET/powershell/
     helloPowershell.ps1"}
     ```

   - Dans le champ Ligne de commande, entrez les paramètres d'exécution du script. Voici un exemple de .

     ```
     helloPowershell.ps1 argument-1 argument-2
     ```

   - (Facultatif) Dans le champ Répertoire de travail, saisissez le nom d'un répertoire de l'instance où vous souhaitez télécharger et exécuter le script.
   - (Facultatif) Dans Execution Timeout (Délai d'exécution), précisez le nombre de secondes durant lesquelles le système doit attendre avant de mettre en échec l'exécution de la commande de script.
6. Dans la section Cibles, identifiez les instances sur lesquelles vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant manuellement les instances ou en indiquant un groupe de ressources.

   Note

   Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.
7. Pour Autres paramètres :
   - Pour Comment (Commentaire), saisissez des informations à propos de cette commande.
   - Pour Délai (secondes), précisez le nombre de secondes durant lesquelles le système doit attendre avant de mettre en échec l'exécution de la commande globale.
8. (Facultatif) Pour Contrôle du débit :
   - Pour Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage d'instances sur lesquelles exécuter simultanément la commande.

     Note

     Si vous avez sélectionné des cibles en indiquant des balises appliquées à des instances gérées ou en indiquant des groupes de ressources AWS, et que vous n'êtes pas sûr du nombre d'instances ciblées, limitez le nombre d'instances pouvant exécuter simultanément le document en spécifiant un pourcentage.
   - Pour Seuil d'erreur, spécifiez quand arrêter l'exécution de la commande sur d'autres instances après son échec sur un nombre ou un pourcentage d'instances. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les

instances sur lesquelles la commande est en cours de traitement peuvent également envoyer des erreurs.

9. (Facultatif) Pour Output options (Options de sortie), pour enregistrer la sortie de la commande dans un fichier, cochez la case Write command output to an S3 bucket (Écrire la sortie de commande vers un compartiment S3). Entrez les noms de compartiment et de préfixe (dossier) dans les zones.

   Note

   Les autorisations S3 qui accordent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour de plus amples informations, veuillez consulter Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). En outre, si le compartiment S3 spécifié se trouve dans un compte AWS différent, assurez-vous que le profil d'instance associé à l'instance dispose des autorisations nécessaires pour écrire dans ce compartiment.

10. Dans la section SNS notifications (Notifications SNS), si vous souhaitez que des notifications sur le statut de l'exécution de la commande soient envoyées, cochez la case Enable SNS notifications (Activer les notifications SNS).

    Pour plus d'informations sur la configuration des notifications Amazon SNS pour la fonctionnalité Exécuter la commande, consultez Surveillance des changements de statut Systems Manager à l'aide de notifications Amazon SNS (p. 1344).

11. Cliquez sur Exécuter.

Exécution d'un PowerShell script depuis à l'Amazon S3aide de l'AWS CLI

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Exécutez la commande suivante .

   Linux

```
aws ssm send-command \
    --document-name "AWS-RunRemoteScript" \
    --targets "Key=instanceids,Values=instance-IDs" \
    --parameters '{"sourceType":["S3"],"sourceInfo":["{\"path\":\"https://
s3.amazonaws.com/path_to_script\"}"],"commandLine":["script_name_and_arguments"]}'
```

   Voici un exemple de .

```
aws ssm send-command \
    --document-name "AWS-RunRemoteScript" \
    --targets "Key=instanceids,Values=i-02573cafcfEXAMPLE" \
    --parameters '{"sourceType":["S3"],"sourceInfo":["{\"path\":
\"https://s3.amazonaws.com/DOC-EXAMPLE-BUCKET/scripts/powershell/
helloWorld.ps1\"}"],"commandLine":["helloWorld.ps1 argument-1 argument-2"]}'
```

   Windows

```
aws ssm send-command ^
    --document-name "AWS-RunRemoteScript" ^
    --targets "Key=instanceids,Values=instance-IDs" ^
    --parameters "sourceType"="S3",sourceInfo='{\"path\":\"https://
s3.amazonaws.com/path_to_script\"}',"commandLine"="script_name_and_arguments"
```

Voici un exemple de .

```
aws ssm send-command ^
    --document-name "AWS-RunRemoteScript" ^
    --targets "Key=instanceids,Values=i-02573cafcfEXAMPLE" ^
    --parameters "sourceType"="S3",sourceInfo='{\"path\":
\"https://s3.amazonaws.com/DOC-EXAMPLE-BUCKET/scripts/powershell/
helloWorld.ps1\"}','commandLine"="helloWorld.ps1 argument-1 argument-2"
```

Référencement des secrets AWS Secrets Manager à partir des paramètres Parameter Store

vous Secrets Manager permet d'organiser et de gérer des données de configuration importantes telles que les informations d'identification, les mots de passe et les clés de licence. Parameter Store est intégré à Secrets Manager afin que vous puissiez récupérer Secrets Manager des secrets lors de l'utilisation d'autres services AWS qui prennent déjà en charge Parameter Store les références aux paramètres. Ces services incluent Amazon EC2, Amazon Elastic Container Service, AWS Lambda, AWS CloudFormation, AWS CodeBuild, AWS CodeDeploy, et d'autres fonctionnalités Systems Manager. Si vous utilisez Parameter Store pour référencer les secrets Secrets Manager, vous créez un processus cohérent et sécurisé permettant d'appeler et d'utiliser les secrets, ainsi que de référencer les données dans votre code et vos scripts de configuration.

Pour plus d'informations sur Secrets Manager, consultez Qu'est-ce qu'AWS Secrets Manager ? dans le Guide de l'utilisateur AWS Secrets Manager.

> **Important**
>
> Parameter Store fonctionne comme un service de passerelle pour les références aux secrets Secrets Manager. Parameter Store ne conserve pas les données ni les métadonnées relatives aux secrets. La référence est sans état.

Restrictions

Notez les restrictions suivantes lorsque vous utilisez Parameter Store pour référencer des secrets Secrets Manager :

- Vous pouvez uniquement récupérer Secrets Manager des secrets en utilisant les actions GetParameter d'API GetParameters et . Les opérations de modification et les actions d'API d'interrogation avancée, telles que DescribeParameters et GetParametersByPath , ne sont pas prises en charge pour Secrets Manager .
- Vous pouvez utiliser l' AWS CLI , l' et l' Outils AWS pour Windows PowerShell pour récupérer un secret à l'aide d' SDKs .Parameter Store
- Lorsque vous récupérez un Secrets Manager secret à partir de Parameter Store , le nom du paramètre doit commencer par le chemin d'accès réservé suivant : /aws/reference/secretsmanager/*secret_ID_in_Secrets_Manager*.

  Voici un exemple : /aws/reference/secretsmanager/CFCreds1
- Parameter Store respecte les stratégies IAM attachées aux secrets Secrets Manager. Par exemple, si l'utilisateur 1 n'a pas accès à Secret A, l'utilisateur 1 ne peut pas récupérer Secret A à l'aide de Parameter Store.
- Les paramètres qui référencent des secrets Secrets Manager ne peuvent pas utiliser les fonctions d'historique ou de gestion des versions Parameter Store.
- Parameter Store respecte les étapes de version Secrets Manager. Si vous référencez une étape de version, elle peut utiliser uniquement des lettres, des chiffres, un point (.), un tiret (-) ou un trait de

soulignement (_). Tous les autres symboles spécifiés dans l'étape de version entraînent l'échec de la référence.

Procédure de référencement d'un secret Secrets Manager avec Parameter Store

La procédure suivante décrit comment référencer un Secrets Manager secret à l'aide d' Parameter Store APIs . La procédure fait référence à d'autres procédures dans le Guide de l'utilisateur AWS Secrets Manager .

> **Note**
>
> Avant de commencer, vérifiez que vous êtes autorisé à référencer des secrets Secrets Manager dans des paramètres Parameter Store. Si vous disposez de privilèges administrateur dans Secrets Manager et Systems Manager , vous pouvez référencer ou récupérer des secrets à l'aide d' Parameter Store APIs . Si vous référencez un Secrets Manager secret dans un Parameter Store paramètre et que vous n'avez pas l'autorisation d'accéder à ce secret, la référence échoue. Pour plus d'informations, consultez Authentification et contrôle d'accès pour AWS Secrets Manager dans le Guide de l'utilisateur AWS Secrets Manager .

Pour référencer un secret Secrets Manager à l'aide de Parameter Store

1. Créez un secret dans Secrets Manager. Pour plus d'informations, consultez Création et gestion des secrets avec AWS Secrets Manager.
2. Référencez un secret à l'aide de l'AWS CLI, des Outils AWS pour Windows PowerShell ou du kit SDK. Lorsque vous référencez un secret Secrets Manager, le nom de paramètre doit commencer par le chemin d'accès réservé suivant : /aws/reference/secretsmanager/. En spécifiant ce chemin, Systems Manager sait comment récupérer le secret à partir de Secrets Manager au lieu de Parameter Store. Voici quelques exemples de paramètres qui référencent correctement des secrets Secrets Manager :

   - /aws/reference/secretsmanager/CFCreds1
   - /aws/reference/secretsmanager/DBPass

   Voici un exemple de code Java qui fait référence à une clé d'accès et à une clé secrète stockées dans Secrets Manager . Cet exemple de code configure un client Amazon DynamoDB. Le code récupère les informations d'identification et les données de configuration à partir de Parameter Store. Les données de configuration sont stockées en tant que paramètre de chaîne dans Parameter Store et les informations d'identification sont stockées dans Secrets Manager. Même si les données de configuration et les informations d'identification sont stockées dans des services distincts, les deux ensembles de données sont accessibles à partir d' à l'Parameter Storeaide de l`GetParameter`'API .

```
/**
* Initialize AWS System Manager Client with default credentials
*/
AWSSimpleSystemsManagement ssm =
 AWSSimpleSystemsManagementClientBuilder.defaultClient();

...

/**
* Example method to launch DynamoDB client with credentials different from default
* @return DynamoDB client
*/
AmazonDynamoDB getDynamoDbClient() {
    //Getting AWS credentials from Secrets manager using GetParameter
    BasicAWSCredentials differentAWSCreds = new BasicAWSCredentials(
            getParameter("/aws/reference/secretsmanager/access-key"),
            getParameter("/aws/reference/secretsmanager/secret-key"));
```

```
    //Initialize the DDB Client with different credentials
    final AmazonDynamoDB client = AmazonDynamoDBClient.builder()
            .withCredentials(new AWSStaticCredentialsProvider(differentAWSCreds))
            .withRegion(getParameter("region")) //Getting config from Parameter Store
            .build();
    return client;
}

/**
* Helper method to retrieve SSM Parameter's value
* @param parameterName identifier of the SSM Parameter
* @return decrypted parameter value
*/
public GetParameterResult getParameter(String parameterName) {
    GetParameterRequest request = new GetParameterRequest();
    request.setName(parameterName);
    request.setWithDecryption(true);
    return ssm.newGetParameterCall().call(request).getParameter().getValue();
}
```

Voici quelques exemples d'AWS CLI.

AWS CLIExemple 1 : Référence à l'aide du nom du secret

Linux

```
aws ssm get-parameter \
    --name /aws/reference/secretsmanager/s1-secret \
    --with-decryption
```

Windows

```
aws ssm get-parameter ^
    --name /aws/reference/secretsmanager/s1-secret ^
    --with-decryption
```

La commande renvoie des informations telles que les suivantes.

```
{
    "Parameter": {
        "Name": "/aws/reference/secretsmanager/s1-secret",
        "Value": "Fl*MEishm!al875",
        "Type": "SecureString",
        "LastModifiedDate": 2018-05-14T21:47:14.743Z,
        "ARN": "arn:aws:secretsmanager:us-east-2:123456789012:secret:s1-secret-
               E18LRP",
        "SourceResult":
              "{
                   \"CreatedDate\": 1526334434.743,
                   \"Name\": \"s1-secret\",
                   \"VersionId\": \"aaabbbccc-1111-222-333-123456789\",
                   \"SecretString\": \"Fl*MEishm!al875\",
                   \"VersionStages\": [\"AWSCURRENT\"],
                   \"ARN\": \"arn:aws:secretsmanager:us-east-2:123456789012:secret:s1-
secret-E18LRP\"
               }"
      }
}
```

AWS CLIExemple 2 : Référence qui inclut l'ID de version

Linux

```
aws ssm get-parameter \
    --name /aws/reference/secretsmanager/s1-secret:11111-aaa-bbb-ccc-123456789 \
    --with-decryption
```

Windows

```
aws ssm get-parameter ^
    --name /aws/reference/secretsmanager/s1-secret:11111-aaa-bbb-ccc-123456789 ^
    --with-decryption
```

La commande renvoie des informations telles que les suivantes.

```
{
    "Parameter": {
        "Name": "/aws/reference/secretsmanager/s1-secret",
        "Value": "Fl*MEishm!al875",
        "Type": "SecureString",
        "LastModifiedDate": 2018-05-14T21:47:14.743Z,
        "ARN": "arn:aws:secretsmanager:us-east-2:123456789012:secret:s1-secret-
               E18LRP",
        "SourceResult":
              "{
                   \"CreatedDate\": 1526334434.743,
                   \"Name\": \"s1-secret\",
                   \"VersionId\": \"11111-aaa-bbb-ccc-123456789\",
                   \"SecretString\": \"Fl*MEishm!al875\",
                   \"VersionStages\": [\"AWSCURRENT\"],
                   \"ARN\": \"arn:aws:secretsmanager:us-east-2:123456789012:secret:s1-
secret-E18LRP\"
               }"
        "Selector": ":11111-aaa-bbb-ccc-123456789"
      }
}
```

AWS CLIExemple 3 : Référence qui inclut l'étape de la version

Linux

```
aws ssm get-parameter \
    --name /aws/reference/secretsmanager/s1-secret:AWSCURRENT \
    --with-decryption
```

Windows

```
aws ssm get-parameter ^
    --name /aws/reference/secretsmanager/s1-secret:AWSCURRENT ^
    --with-decryption
```

La commande renvoie des informations telles que les suivantes.

```
{
    "Parameter": {
```

```
        "Name": "/aws/reference/secretsmanager/s1-secret",
        "Value": "Fl*MEishm!al875",
        "Type": "SecureString",
        "LastModifiedDate": 2018-05-14T21:47:14.743Z,
        "ARN": "arn:aws:secretsmanager:us-east-2:123456789012:secret:s1-secret-
                E18LRP",
        "SourceResult":
              "{
                   \"CreatedDate\": 1526334434.743,
                   \"Name\": \"s1-secret\",
                   \"VersionId\": \"11111-aaa-bbb-ccc-123456789\",
                   \"SecretString\": \"Fl*MEishm!al875\",
                   \"VersionStages\": [\"AWSCURRENT\"],
                   \"ARN\": \"arn:aws:secretsmanager:us-east-2:123456789012:secret:s1-
secret-E18LRP\"
                }"
        "Selector": ":AWSCURRENT"
      }
}
```

Intégration à d'autres produits et services

Systems Managerdispose d'une intégration embarquée pour les produits et services suivants.

Ansible	Ansible est une plateforme d'automatisation informatique qui facilite le déploiement de vos applications et systèmes. Systems Managerfournit le `AWS-ApplyAnsiblePlaybooks` SSM document qui vous permet de créer State Manager des associations qui exécutent des manuels stratégiques Ansible. En savoir plus Création d'associations exécutant des manuels stratégiques Ansible (p. 1030)
Chef	Chef est un outil d'automatisation informatique qui facilite le déploiement de vos applications et systèmes. Systems Managerfournit le `AWS-ApplyChefRecipes` SSM document, qui vous permet de créer State Manager des associations qui exécutent des recettes Chef. En savoir plus Création d'associations qui exécutent des recettes de Chef (p. 1035) s'intègre Systems Manager également aux InSpec profils Chef, ce qui vous permet d'exécuter des analyses de conformité et d'afficher les instances conformes et non conformes.

	En savoir plus Utilisation des InSpec profils Chef avec Systems Manager la conformité (p. 145)
GitHub	GitHubpermet l'hébergement pour le contrôle de version et la collaboration au développement logiciel. Systems Managerfournit le `AWS-RunDocument` , qui vous permet d'exécuter SSM des documents stockés dans GitHub , et le `AWS-RunRemoteScript` SSM document , qui vous permet d'exécuter des scripts stockés dans GitHub . En savoir plus • Exécution de documents Systems Manager de commande à partir d'emplacements distants (p. 1286) • Exécution de scripts depuis GitHub (p. 139)
Jenkins	Jenkins est un serveur d'automatisation open source qui permet aux développeurs de générer, tester et déployer leurs logiciels en toute fiabilité. Systems ManagerL'automatisation peut être utilisée comme étape après génération pour préinstaller des versions d'application dans des Amazon Machine Images (AMI). En savoir plus Procédure : Utilisation d'Automation avec Jenkins (p. 667)

Rubriques

• Exécution de scripts depuis GitHub (p. 139)

Exécution de scripts depuis GitHub

Cette rubrique décrit comment utiliser le `AWS-RunRemoteScript` document SSM prédéfini pour télécharger des scripts à partir d' GitHub , y compris les playbooks Ansible, Python, Ruby et PowerShell les scripts . Grâce à ce document, vous n'avez plus besoin de porter manuellement des scripts dans Amazon EC2 ou de les encapsuler dans SSM des documents. Systems Manager L'intégration à GitHub promeut l'infrastructure sous forme de code, ce qui réduit le temps nécessaire à la gestion des instances tout en normalisant les configurations dans votre parc.

Vous pouvez également créer des documents SSM personnalisés qui vous permettent de télécharger et d'exécuter des scripts ou d'autres documents SSM à partir d'emplacements distants. Pour plus d'informations, consultez la section Création de documents composites (p. 1273).

Vous pouvez également télécharger un répertoire contenant plusieurs scripts. Lorsque vous exécutez le script principal dans le répertoire, exécute Systems Manager également tous les scripts référencés qui sont inclus dans le répertoire.

Notez les informations importantes suivantes relatives à l'exécution de scripts à partir de GitHub.

- Systems Managerne vérifie pas que votre script est capable de s'exécuter sur une instance. Avant de télécharger et d'exécuter le script, vous devez vérifier que le logiciel requis est installé sur l'instance. Vous pouvez également créer un document composite qui installe le logiciel par l'intermédiaire de Fonctionnalité Exécuter la commande ou de State Manager, puis télécharge et exécute le script.
- Il vous incombe de vous assurer que toutes les GitHub exigences sont satisfaites. Cela inclut l'actualisation de votre jeton d'accès, si nécessaire. Vous devez également vous assurer que vous ne dépassez pas le nombre de requêtes authentifiées ou non authentifiées. Pour de plus amples informations, veuillez consulter la documentation GitHub.

Rubriques

- Exécuter des manuels stratégiques Ansible à partir deGitHub (p. 140)
- Exécuter des scripts Python depuisGitHub (p. 142)

Exécuter des manuels stratégiques Ansible à partir deGitHub

Cette section inclut les procédures vous permettant d'exécuter des manuels stratégiques Ansible à partir d' à l'aide de la console ou de l' GitHub .AWS CLI

Avant de commencer

Si vous prévoyez d'exécuter un script stocké dans un GitHub référentiel privé, vous devez créer un Systems Manager `SecureString` paramètre pour votre jeton d'accès GitHub de sécurité. Vous ne pouvez pas accéder à un script dans un GitHub référentiel privé en transmettant manuellement votre jeton via SSH. Le jeton d'accès doit être transmis en tant que paramètre Systems Manager `SecureString`. Pour plus d'informations sur la création d'un paramètre `SecureString`, consultez Création de paramètres Systems Manager (p. 243).

Exécution d'un manuel stratégique Ansible à partir dGitHub' (console)

Exécution d'un manuel stratégique Ansible à partir d'GitHub

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fonctionnalité Exécuter la commande.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Fonctionnalité Exécuter la commande.
3. Choisissez Exécuter une commande.
4. Dans la liste Document de commande, choisissez AWS-RunRemoteScript.
5. Dans Paramètres de la commande, procédez comme suit :

   - Dans Type de source, sélectionnez GitHub .
   - Dans la zone Source Info (Informations sur la source), entrez les informations requises pour accéder à la source en respectant le format suivant :

```
{
    "owner": "owner_name",
    "repository": "repository_name",
    "branch": "branch_name",
    "path": "path_to_scripts_or_directory",
    "tokenInfo": "{{ssm-secure:SecureString_parameter_name}}"
```

```
}
```

Cet exemple télécharge un fichier nommé `webserver.yml`.

```
{
    "owner": "TestUser1",
    "repository": "GitHubPrivateTest",
    "branch": "myBranch",
    "path": "scripts/webserver.yml",
    "tokenInfo": "{{ssm-secure:mySecureStringParameter}}"
}
```

> **Note**
>
> `"branch"` n'est requis que si votre document SSM est stocké dans une branche autre que `master`.
> Pour utiliser la version de vos scripts qui se trouvent dans une validation particulière dans votre référentiel, utilisez `commitID` avec `getOptions` au lieu d' `branch` . Par exemple : `"getOptions": "commitID:bbc1ddb94...b76d3bEXAMPLE",`

- Dans le champ Ligne de commande, entrez les paramètres d'exécution du script. Voici un exemple de .

  `ansible-playbook -i “localhost,” --check -c local webserver.yml`
- (Facultatif) Dans le champ Répertoire de travail, saisissez le nom d'un répertoire de l'instance où vous souhaitez télécharger et exécuter le script.
- (Facultatif) Dans Execution Timeout (Délai d'exécution), précisez le nombre de secondes durant lesquelles le système doit attendre avant de mettre en échec l'exécution de la commande de script.

6. Dans la section Cibles, identifiez les instances sur lesquelles vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant manuellement les instances ou en indiquant un groupe de ressources.

   > **Note**
   >
   > Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.

7. Pour Autres paramètres :

   - Pour Comment (Commentaire), saisissez des informations à propos de cette commande.
   - Pour Délai (secondes), précisez le nombre de secondes durant lesquelles le système doit attendre avant de mettre en échec l'exécution de la commande globale.

8. (Facultatif) Pour Contrôle du débit :

   - Pour Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage d'instances sur lesquelles exécuter simultanément la commande.

     > **Note**
     >
     > Si vous avez sélectionné des cibles en indiquant des balises appliquées à des instances gérées ou en indiquant des groupes de ressources AWS, et que vous n'êtes pas sûr du nombre d'instances ciblées, limitez le nombre d'instances pouvant exécuter simultanément le document en spécifiant un pourcentage.
   - Pour Seuil d'erreur, spécifiez quand arrêter l'exécution de la commande sur d'autres instances après son échec sur un nombre ou un pourcentage d'instances. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les instances sur lesquelles la commande est en cours de traitement peuvent également envoyer des erreurs.

9. (Facultatif) Pour Output options (Options de sortie), pour enregistrer la sortie de la commande dans un fichier, cochez la case Write command output to an S3 bucket (Écrire la sortie de commande vers un compartiment S3). Entrez les noms de compartiment et de préfixe (dossier) dans les zones.

   Note

   Les autorisations S3 qui accordent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour de plus amples informations, veuillez consulter Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). En outre, si le compartiment S3 spécifié se trouve dans un compte AWS différent, assurez-vous que le profil d'instance associé à l'instance dispose des autorisations nécessaires pour écrire dans ce compartiment.

10. Dans la section SNS notifications (Notifications SNS), si vous souhaitez que des notifications sur le statut de l'exécution de la commande soient envoyées, cochez la case Enable SNS notifications (Activer les notifications SNS).

    Pour plus d'informations sur la configuration des notifications Amazon SNS pour la fonctionnalité Exécuter la commande, consultez Surveillance des changements de statut Systems Manager à l'aide de notifications Amazon SNS (p. 1344).

11. Cliquez sur Exécuter.

Exécution d'un manuel stratégique Ansible à partir de à l'aide de l'GitHubAWS CLI

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Exécutez la commande suivante pour télécharger et exécuter un script à partir d' GitHub .

```
aws ssm send-command \
    --document-name "AWS-RunRemoteScript" \
    --instance-ids "instance-IDs"\
     --parameters '{"sourceType":["GitHub"],"sourceInfo":
["{\"owner\":\"owner_name\", \"repository\": \"repository_name\",
 \"path\": \"path_to_file_or_directory\", \"tokenInfo\":\"{{ssm-
secure:name_of_your_SecureString_parameter}}\" }"],"commandLine":["commands_to_run"]}'
```

   Voici un exemple de commande à exécuter sur une machine Linux locale.

```
aws ssm send-command \
    --document-name "AWS-RunRemoteScript" \
    --instance-ids "i-02573cafcfEXAMPLE" \
    --parameters '{"sourceType":["GitHub"],"sourceInfo":["{\"owner\":\"TestUser1\",
 \"repository\": \"GitHubPrivateTest\", \"path\": \"scripts/webserver.yml\",
 \"tokenInfo\":\"{{ssm-secure:mySecureStringParameter}}\" }"],"commandLine":["ansible-
playbook -i “localhost,” --check -c local webserver.yml"]}'
```

Exécuter des scripts Python depuisGitHub

Cette section inclut les procédures vous permettant d'exécuter des scripts Python à partir d' à l'aide de la console ou de l' GitHub .AWS CLI

Exécution d'un script Python à partir dGitHub' (console)

Exécuter un script Python depuisGitHub

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fonctionnalité Exécuter la commande.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Fonctionnalité Exécuter la commande.
3. Choisissez Exécuter une commande.
4. Dans la liste Document de commande, choisissez AWS-RunRemoteScript.
5. Dans Command parameters (Paramètres de la commande), procédez comme suit :

   - Dans Type de source, sélectionnez GitHub .
   - Dans la zone Source Info (Informations sur la source), entrez les informations requises pour accéder à la source en respectant le format suivant :

     ```
     {
         "owner": "owner_name",
         "repository": "repository_name",
         "branch": "branch_name",
         "path": "path_to_document",
         "tokenInfo": "{{ssm-secure:SecureString_parameter_name}}"
     }
     ```

     L'exemple suivant télécharge un répertoire de scripts nommé complex-script.

     ```
     {
         "owner": "TestUser1",
         "repository": "SSMTestDocsRepo",
         "branch": "myBranch",
         "path": "scripts/python/complex-script",
         "tokenInfo": "{{ssm-secure:myAccessTokenParam}}"
     }
     ```

     Note

     `"branch"` n'est requis que si vos scripts sont stockés dans une branche autre que `master`.
     Pour utiliser la version de vos scripts qui se trouvent dans une validation particulière dans votre référentiel, utilisez `commitID` avec `getOptions` au lieu d' `branch` . Par exemple : `"getOptions": "commitID:bbc1ddb94...b76d3bEXAMPLE",`
   - Dans le champ Ligne de commande, entrez les paramètres pour l'exécution du script. Voici un exemple de .

     ```
     mainFile.py argument-1 argument-2
     ```

     Cet exemple exécute `mainFile.py` , qui peut ensuite exécuter d'autres scripts dans le `complex-script` répertoire.
   - (Facultatif) Dans le champ Répertoire de travail, tapez le nom d'un répertoire de l'instance dans lequel vous souhaitez télécharger et exécuter le script.

- (Facultatif) Pour Délai d'exécution, précisez le nombre de secondes durant lesquelles le système doit attendre avant de mettre en échec l'exécution de la commande de script.

6. Dans la section Cibles, identifiez les instances sur lesquelles vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant manuellement les instances ou en indiquant un groupe de ressources.

   Note

   Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.

7. Pour Autres paramètres :

   - Pour Comment (Commentaire), saisissez des informations à propos de cette commande.
   - Pour Délai (secondes), précisez le nombre de secondes durant lesquelles le système doit attendre avant de mettre en échec l'exécution de la commande globale.

8. (Facultatif) Pour Contrôle du débit :

   - Pour Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage d'instances sur lesquelles exécuter simultanément la commande.

     Note

     Si vous avez sélectionné des cibles en indiquant des balises appliquées à des instances gérées ou en indiquant des groupes de ressources AWS, et que vous n'êtes pas sûr du nombre d'instances ciblées, limitez le nombre d'instances pouvant exécuter simultanément le document en spécifiant un pourcentage.

   - Pour Seuil d'erreur, spécifiez quand arrêter l'exécution de la commande sur d'autres instances après son échec sur un nombre ou un pourcentage d'instances. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les instances sur lesquelles la commande est en cours de traitement peuvent également envoyer des erreurs.

9. (Facultatif) Pour Output options (Options de sortie), pour enregistrer la sortie de la commande dans un fichier, cochez la case Write command output to an S3 bucket (Écrire la sortie de commande vers un compartiment S3). Entrez les noms de compartiment et de préfixe (dossier) dans les zones.

   Note

   Les autorisations S3 qui accordent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour de plus amples informations, veuillez consulter Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). En outre, si le compartiment S3 spécifié se trouve dans un compte AWS différent, assurez-vous que le profil d'instance associé à l'instance dispose des autorisations nécessaires pour écrire dans ce compartiment.

10. Dans la section SNS notifications (Notifications SNS), si vous souhaitez que des notifications sur le statut de l'exécution de la commande soient envoyées, cochez la case Enable SNS notifications (Activer les notifications SNS).

    Pour plus d'informations sur la configuration des notifications Amazon SNS pour la fonctionnalité Exécuter la commande, consultez Surveillance des changements de statut Systems Manager à l'aide de notifications Amazon SNS (p. 1344).

11. Cliquez sur Exécuter.

Exécuter un script Python depuis à l'GitHubaide de l'AWS CLI

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Exécutez la commande suivante pour télécharger et exécuter un script à partir d' GitHub .

```
aws ssm send-command --document-name "AWS-RunRemoteScript" --instance-
ids "instance-IDs" --parameters '{"sourceType":["GitHub"],"sourceInfo":
["{\"owner\":\"owner_name\", \"repository\":\"repository_name\", \"path\":
 \"path_to_script_or_directory"}"],"commandLine":["commands_to_run"]}'
```

Voici un exemple de .

```
aws ssm send-command --document-name "AWS-RunRemoteScript" --instance-ids
 "i-02573cafcfEXAMPLE" --parameters '{"sourceType":["GitHub"],"sourceInfo":["{\"owner
\":\"TestUser1\", \"repository\":\"GitHubTestPublic\", \"path\": \"scripts/python/
complex-script\"}"],"commandLine":["mainFile.py argument-1 argument-2 "]}'
```

Cet exemple télécharge un répertoire de scripts appelé `complex-script` . L`commandLine`'entrée exécute `mainFile.py` , qui peut ensuite exécuter d'autres scripts dans le `complex-script` répertoire.

Utilisation des InSpec profils Chef avec Systems Manager la conformité

Systems Managers'intègre à Chef InSpec . InSpec est un framework de test open source qui vous permet de créer des profils lisibles par l'utilisateur à stocker dans GitHub ou Amazon S3 . Ensuite, vous pouvez utiliser Systems Manager pour exécuter des analyses de conformité et afficher les instances conformes et non conformes. Un profil est une exigence en termes de sécurité, de conformité ou de stratégie pour votre environnement informatique. Par exemple, vous pouvez créer des profils qui effectuent les vérifications suivantes lorsque vous analysez vos instances avec la conformité Systems Manager :

- Vérifier si des ports spécifiques sont ouverts ou fermés.
- Vérifier si des applications spécifiques sont en cours d'exécution.
- Vérifier si certains packages sont installés.
- Vérifier les clés de registre Windows pour des propriétés spécifiques.

Vous pouvez créer InSpec des profils pour les instances EC2 et les serveurs sur site ou machines virtuelles (VM) que vous gérez avec Systems Manager . L'exemple de InSpec profil Chef suivant vérifie si le port 22 est ouvert.

```
control 'Scan Port' do
impact 10.0
title 'Server: Configure the service port'
desc 'Always specify which port the SSH server should listen to.
Prevent unexpected settings.'
describe sshd_config do
its('Port') { should eq('22') }
end
end
```

InSpecinclut un ensemble de ressources qui vous aident à écrire rapidement des contrôles de vérification et d'audit. InSpecutilise le langage spécifique au InSpec domaine (DSL) pour écrire ces contrôles en

Ruby. Vous pouvez également utiliser les profils créés par une grande communauté d'InSpecutilisateurs. Par exemple, le projet DevSec chef-os-hardening sur GitHub inclut des dizaines de profils pour vous aider à sécuriser vos instances et serveurs. Vous pouvez créer et stocker des profils dans GitHub ou Amazon Simple Storage Service (Amazon S3).

Fonctionnement

Voici comment utiliser les InSpec profils avec Systems Manager la conformité .

1. Identifiez les InSpec profils prédéfinis que vous souhaitez utiliser ou créez les vôtres. Vous pouvez utiliser des profils prédéfinis sur GitHub pour commencer. Pour plus d'informations sur la création de vos propres InSpec profils, consultez ProfilInSpecs Chef.
2. Stockez les profils dans un GitHub référentiel public ou privé, ou dans un compartiment S3.
3. Exécutez la vérification de conformité avec vos InSpec profils à l'aide du SSM document AWS-RunInspecChecks. Vous pouvez lancer une analyse de conformité en utilisant Fonctionnalité Exécuter la commande (pour les analyses à la demande), ou programmer des analyses de conformité régulières à l'aide de State Manager.
4. Identifiez les instances non conformes à l'aide de l'API de conformité ou de la console de conformité Systems Manager.

   Note

   Chef utilise un client sur vos instances pour traiter le profil. Vous n'avez pas besoin d'installer le client. Lorsque Systems Manager exécute le document SSM AWS-RunInspecChecks, le système vérifie si le client est installé. Si ce n'est pas le cas, Systems Manager installe le client Chef lors de l'analyse, puis le désinstalle une fois l'analyse terminée.

Exécution d'une analyse InSpec de conformité

Cette section inclut des informations sur la façon d'exécuter une analyse InSpec de conformité à l'aide de la Systems Manager console et de l' AWS CLI . La procédure pour la console montre comment configurer State Manager pour exécuter l'analyse. La AWS CLI procédure montre comment configurer Fonctionnalité Exécuter la commande pour exécuter l'analyse.

Exécution d'une analyse InSpec de conformité avec à l'State Manageraide de la console

Pour exécuter une analyse InSpec de conformité avec à l'aide de la State Manager consoleAWS Systems Manager

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez State Manager.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez State Manager.
3. Choisissez Créer une association.
4. Dans la section Fournir les détails de l'association, entrez un nom.
5. Dans la liste Document, choisissez AWS-RunInspecChecks.

6. Dans la liste Version du document, choisissez Dernière version lors de l'exécution.
7. Dans la section Parameters (Paramètres), dans la liste Source Type (Type de source), choisissez GitHub ou S3.

   Si vous choisissez GitHub , saisissez le chemin d'accès à un InSpec profil dans un GitHub référentiel public ou privé dans le champ Source Info (Informations sur la source). Voici un exemple de chemin d'accès à un profil public fourni par l'équipe Systems Manager à partir de l'emplacement suivant : https://github.com/awslabs/amazon-ssm/tree/master/Compliance/InSpec/PortCheck.

   ```
   {"owner":"awslabs","repository":"amazon-ssm","path":"Compliance/InSpec/
   PortCheck","getOptions":"branch:master"}
   ```

   Si vous choisissez S3, saisissez une URL valide renvoyant à un profil dans un compartiment S3 dans le InSpec champ Source Info (Informations sur la source).

   Pour plus d'informations sur l'Systems Managerintégration d' avec GitHub et Amazon S3 , consultez Exécution de scripts depuis GitHub (p. 139) .
8. Dans la section Cibles, identifiez les instances sur lesquelles vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant manuellement les instances ou en indiquant un groupe de ressources.

   Note

   Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.
9. Dans la section Specify schedule (Spécifier la planification), utilisez les options du générateur de planification pour créer une planification qui spécifie à quel moment vous souhaitez que l'analyse de conformité s'exécute.
10. (Facultatif) Pour Contrôle du débit :

    - Pour Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage d'instances sur lesquelles exécuter simultanément la commande.

      Note

      Si vous avez sélectionné des cibles en indiquant des balises appliquées à des instances gérées ou en indiquant des groupes de ressources AWS, et que vous n'êtes pas sûr du nombre d'instances ciblées, limitez le nombre d'instances pouvant exécuter simultanément le document en spécifiant un pourcentage.
    - Pour Seuil d'erreur, spécifiez quand arrêter l'exécution de la commande sur d'autres instances après son échec sur un nombre ou un pourcentage d'instances. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les instances sur lesquelles la commande est en cours de traitement peuvent également envoyer des erreurs.
11. (Facultatif) Pour Output options (Options de sortie), pour enregistrer la sortie de la commande dans un fichier, cochez la case Write command output to an S3 bucket (Écrire la sortie de commande vers un compartiment S3). Entrez les noms de compartiment et de préfixe (dossier) dans les zones.

    Note

    Les autorisations S3 qui accordent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour de plus amples informations, veuillez consulter Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). En outre, si le compartiment S3 spécifié se trouve dans un compte AWS différent, assurez-vous que le profil d'instance associé à l'instance dispose des autorisations nécessaires pour écrire dans ce compartiment.

12. Choisissez Create Association (Créer une association). Le système crée l'association et exécute automatiquement l'analyse de conformité.
13. Patientez quelques minutes pendant que l'analyse s'effectue, puis choisissez Conformité dans le panneau de navigation.
14. Dans Instances gérées correspondantes, recherchez des instances pour lesquelles la colonne Type de conformité contient la valeur Custom:Inspec.
15. Choisissez un ID d'instance pour afficher le détail des statuts non conformes.

Exécution d'une analyse InSpec de conformité avec à l'Fonctionnalité Exécuter la commandeaide de l'AWS CLI

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Exécutez l'une des commandes suivantes pour exécuter un InSpec profil à partir d' GitHub ou d' Amazon S3 .

   La commande utilise les paramètres suivants :

   - sourceType: GitHub ou Amazon S3
   - sourceInfo: URL vers le dossier de InSpec profil dans GitHub ou un compartiment S3. Le dossier doit contenir le InSpec fichier de base (*.yml) et tous les contrôles associés (*.rb).

   GitHub

```
aws ssm send-command --document-name "AWS-RunInspecChecks" --targets
 '[{"Key":"tag:tag_name","Values":["tag_value"]}]' --parameters '{"sourceType":
["GitHub"],"sourceInfo":["{\"owner\":\"owner_name\", \"repository\":
\"repository_name\", \"path\": \"Inspec.yml_file"}"]}'
```

   Voici un exemple de .

```
aws ssm send-command --document-name "AWS-RunInspecChecks" --targets
 '[{"Key":"tag:testEnvironment","Values":["webServers"]}]' --parameters '{"sourceType":
["GitHub"],"getOptions":"branch:master","sourceInfo":["{\"owner\":\"awslabs\",
 \"repository\":\"amazon-ssm\", \"path\": \"Compliance/InSpec/PortCheck\"}"]}'
```

   Amazon S3

```
aws ssm send-command --document-name "AWS-RunInspecChecks" --targets
 '[{"Key":"tag:tag_name","Values":["tag_value"]}]' --parameters'{"sourceType":
["S3"],"sourceInfo":["{\"path\":\"https://s3.amazonaws.com/DOC-EXAMPLE-
BUCKET/Inspec.yml_file\"}"]}'
```

   Voici un exemple de .

```
aws ssm send-command --document-name "AWS-RunInspecChecks" --targets
 '[{"Key":"tag:testEnvironment","Values":["webServers"]}]' --parameters'{"sourceType":
["S3"],"sourceInfo":["{\"path\":\"https://s3.amazonaws.com/DOC-EXAMPLE-BUCKET/InSpec/
PortCheck.yml\"}"]}'
```

3. Exécutez la commande suivante pour afficher un récapitulatif de l'analyse de conformité.

```
aws ssm list-resource-compliance-summaries --filters
 Key=ComplianceType,Values=Custom:Inspec
```

4. Exécutez la commande suivante pour afficher les détails d'une instance qui n'est pas conforme.

```
aws ssm list-compliance-items --resource-ids instance_ID --resource-type
 ManagedInstance --filters Key=DocumentName,Values=AWS-RunInspecChecks
```

Exemples d'intégration issus de la communauté

Les sections suivantes fournissent des liens vers des billets de blogs, des articles et des exemples fournis par la communauté.

> **Note**
>
> Ces liens sont fournis à titre informatif uniquement et ne doivent pas être considérés comme une liste exhaustive ni comme une approbation du contenu des exemples. n'AWSest pas responsable du contenu ou de l'exactitude de ce contenu.

Billets de blogs

Application Management

- Guide complet d'utilisation du magasin de paramètres AWS Systems Manager

  Sean Ziegler offre un aperçu concis des fonctionnalités Parameter Store et fournit un exemple Boto3 pour interagir avec Parameter Store.

  Publié en mai 2020
- Gardez vos secrets en sécurité avec AWS Systems Manager Parameter Store et Node

  Amir Boroumand explique comment enregistrer et récupérer un mot de passe à l'aide de l' Parameter Store avec Node.

  Publié en octobre 2019
- Utilisation de paramètres SSM avec des CloudFormation modèles et des projets Terraform

  J Cole Morrison montre comment créer des paramètres à utiliser dans les modèles AWS CloudFormation et les projets Terraform, pour vous permettre de référencer des valeurs avec moins d'erreurs humaines.

  Publié en août 2019
- Utilisation de l'AWS Systems Manager Parameter Store pour la configuration

  Découvrez comment stocker et récupérer les paramètres de configuration d'une application lors de son exécution au lieu de coder en dur les valeurs de configuration dans le code et les fichiers de configuration d'un exemple d'application. L'exemple d'application, une simple application console .NET Core, montre comment utiliser le kit SDK AWS pour .NET pour récupérer les valeurs de configuration à partir des Parameter Store.

  Publié en mars 2019
- Utilisation de AWS Systems Manager Parameter Store SecureString paramètres dans AWS CloudFormation les modèles

  Apprenez à utiliser le texte brut et `SecureString` les paramètres dans vos AWS CloudFormation modèles grâce à l'utilisation de références dynamiques pour extraire les valeurs des paramètres.

Publié en octobre 2018

- Utiliser les étiquettes de paramètre pour faciliter la mise à jour de la configuration entre les environnements

  Découvrez comment utiliser une étiquette de paramètre comme alias pour une version de paramètre. Vous pouvez regrouper les versions de paramètres entre les hiérarchies à l'aide d'étiquettes et déployer de nouvelles mises à jour de paramètres dans tous les environnements à l'aide de hiérarchies et d'étiquettes.

  Publié en juillet 2018

- Intégration àAWS CloudFormation AWS Systems Manager Parameter Store

  Découvrez comment utiliser les paramètres Parameter Store de vos modèles AWS CloudFormation pour simplifier les mises à jour de pile incluant des paramètres et assurer la cohérence en utilisant les valeurs stockées dans Parameter Store. Grâce à cette intégration, votre code reste intact pendant que l'opération de mise à jour de pile récupère automatiquement la dernière valeur de paramètre.

  Publié en décembre 2017

- La bonne façon de stocker des secrets à l'aide d' Parameter Store

  Evan Johnson présente une étude de cas sur la façon dont Segment gère de manière centralisée et sécurisée ses secrets à l'aide d'une combinaison de Parameter Store, beaucoup de code Terraform, une chambre, en mettant l'accent sur la mise en place et le fonctionnement en production de Parameter Store.

  Publié en août 2017

- Joindre un domaine Microsoft Active Directory avec Parameter Store et Amazon EC2 Systems Manager Documents

  Découvrez un scénario de gestion centralisée de la configuration, avec un exemple de connexion d'instances EC2 à un Microsoft Active Directory. Ce billet vous montre comment lancer une instance EC2 qui consomme et utilise des valeurs de configuration stockées en tant que paramètres Parameter Store pour vous connecter à votre domaine Active Directory.

  Publié en juin 2017

- Utilisez Parameter Store pour accéder en toute sécurité aux secrets et aux données de configuration dans AWS CodeDeploy

  Découvrez comment simplifier vos flux de travail CodeDeploy en utilisant Parameter Store pour stocker et référencer un secret de configuration. Cela permet non seulement d'améliorer votre posture de sécurité, mais aussi d'automatiser votre déploiement, car vous n'avez pas à modifier manuellement les données de configuration dans votre code source.

  Publié en mars 2017

- Secrets dans AWS

  Stephen Price décrit comment vous pouvez utiliser Parameter Store pour gérer et utiliser des secrets dans votre langage de programmation préféré, pour gérer les secrets d'architectures basées sur le cloud, notamment les microservices ou les applications conteneurisées.

  Publié en mars 2017

- Utilisation d' Parameter Store avec AWS CodePipeline

  Il McElhattan montre comment utiliser efficacement dans le cadre d'un pipeline de livraison continue à l'aide d' Parameter Store .AWS CodePipeline

  Publié en mars 2017

- Gestion des secrets pour les applications Amazon ECS à l'aide Parameter Store d' et IAM des rôles pour les tâches

  En utilisant Parameter Store les IAM rôles de tâche et , vous pouvez créer un magasin central de gestion des secrets et une couche d'accès bien intégrée qui permet aux applications d'accéder uniquement aux clés dont elles ont besoin, de restreindre l'accès sur la base d'un conteneur et de chiffrer davantage les secrets avec des clés KMS personnalisées.

  Publié en janvier 2017

Actions et modifications

- Fourniture d'autorisations d'instance temporaires avec AWS Systems Manager Automations

  Découvrez comment fournir des autorisations temporaires aux Amazon EC2 instances au sein de vos documents Automation. Cela vous aide à éviter de modifier les profils d'instance attachés aux instances à long terme et contenant uniquement les autorisations de base requises pour les fonctionnalités Systems Manager.

  Publié en décembre 2019
- Automatisation du cloud : Sécurité assurée efficacement par AWS

  Josh Frantz, responsable de la sécurité des logiciels chez Rapid7, montre comment automatiser les tâches courantes afin que vous puissiez sécuriser plus efficacement votre environnement AWS et vous concentrer sur la résolution de problèmes importants, engageants et difficiles.

  Publié en août 2019
- Gestion des ressources AWS dans plusieurs comptes et régions à l'aide d'AWS Systems ManagerAutomation

  Découvrez comment gérer vos ressources dans plusieurs comptes et régions AWS à l'aide d'Systems ManagerAutomation.

  Publié en janvier 2019
- Onica présente les utilisations pour les nouvelles actions d'automatisation AWS Systems Manager

  Onica utilise des exemples concrets représentant certains des problèmes qu'ils voient sur le terrain tout en aidant les clients AWS à démontrer comment les actions Automation peuvent être utilisées pour résoudre ces problèmes.

  Publié en août 2018

Instances et nœuds

- Comment corriger les instances Amazon EC2 Windows dans des sous-réseaux privés à l'aide d' AWS Systems Manager

  Découvrez comment corriger les instances Amazon EC2 Windows dans des sous-réseaux privés sans connectivité Internet.

  Publié en décembre 2018
- Des correctifs centralisés sur plusieurs comptes et plusieurs régions avec AWS Systems Manager Automation

  Découvrez comment utiliser Systems Manager Automation pour corriger vos instances gérées sur plusieurs comptes et régions AWS.

Publié en novembre 2018

- Correctifs multiplateformes évolutifs avec AWS Systems Manager

  Apprenez à corriger Amazon EC2 les instances à grande échelle avec Systems Manager .

  Publié en avril 2018

Ressources partagées

- Écriture de vos propres AWS Systems Manager documents

  Découvrez comment créer vos propres Systems Manager documents.

  Publié en mai 2018

Gestion des opérations

Gestion des opérations est un ensemble de fonctionnalités qui vous aident à gérer vos ressources AWS.

Rubriques

- AWS Systems Manager Explorer (p. 153)
- AWS Systems Manager OpsCenter (p. 176)
- Tableaux de bord Amazon CloudWatch hébergés par Systems Manager (p. 210)
- Tableaux de bord Personal Health et Trusted Advisor hébergés par Systems Manager (p. 210)

AWS Systems Manager Explorer

AWS Systems Manager Explorer est un tableau de bord des opérations personnalisable qui présente des informations sur vos ressources AWS. Explorer affiche une vue agrégée des données d'opérations (OpsData) pour vos comptes et pour toutes les régions AWS. Dans Explorer , OpsData inclut les métadonnées relatives à vos instances EC2, les détails de conformité des correctifs et les détails de conformité des State Manager associations. inclut OpsData également des informations sur la prise en charge des services AWS tels qu'AWS Trusted Advisor, AWS Compute Optimizer et sur vos cas AWS Support.

Pour lever une garantie opérationnelle, affiche Explorer également des éléments de travail opérationnels (OpsItems). Explorer Fournit un contexte sur la façon dont OpsItems sont répartis entre vos unités commerciales ou applications, leur évolution au fil du temps et leur variation par catégorie. Vous pouvez regrouper et filtrer les informations dans Explorer pour vous concentrer sur les éléments qui vous intéressent et qui nécessitent une action. Lorsque vous identifiez des problèmes prioritaires, vous pouvez utiliser Systems Manager OpsCenter pour exécuter des runbooks Automation et résoudre rapidement ces problèmes.

L'image suivante montre certaines des zones de rapport individuelles, appelées widgets, qui sont disponibles dans Explorer.

Quelles sont les fonctions d'Explorer ?

Explorer comprend les fonctions suivantes :

- Affichage personnalisable des informations exploitables : Explorer comprend des widgets à glisser-déposer qui affichent automatiquement des informations exploitables concernant vos ressources AWS. Explorer affiche des informations dans deux types de widgets.
  - Widgets d'information : Ces widgets résument les données provenant d' Amazon EC2 , Systems Manager Gestionnaire de correctifs , et des services AWS pris en charge, tels qu'AWS Trusted Advisor, AWS Compute Optimizer et AWS Support.Systems ManagerState Manager Ces widgets fournissent un contexte important pour vous aider à comprendre l'état et les risques opérationnels de vos ressources AWS. Les exemples de widgets d'information incluent le nombre d'instances, l'instance par AMI, les instances non conformes pour l'application de correctifs, les associations non conformes et les cas de Centre de support.
  - OpsItemwidgets : Un Systems Manager OpsItem est un élément de travail opérationnel lié à une ou plusieurs ressources AWS. OpsItems Les sont une fonction d' Systems Manager OpsCenter peuvent nécessiter que OpsItems des ingénieurs étudient et éventuellement corrigent un problème.DevOps Les OpsItems peuvent, par exemple, être l'utilisation élevée par une instance EC2 du processeur, les volumes Amazon Elastic Block Store détachés, un échec du déploiement AWS CodeDeploy ou un échec d'exécution Systems Manager Automation. Parmi les exemples de OpsItem widgets, citons Open OpsItem summary, by status et OpsItem over time.OpsItems
- Filtres : Chaque widget offre la possibilité de filtrer les informations en fonction du compte, de la région et de la balise AWS. Les filtres vous aident à affiner rapidement les informations affichées dans Explorer.
- Liens directs vers les écrans de service : Pour vous aider à analyser les problèmes liés aux ressources AWS, les Explorer widgets contiennent des liens directs vers les écrans de service associés. Les filtres appliqués à un widget restent actifs si vous accédez à un écran de service associé.

- Groupes : Pour vous aider à comprendre les types de problèmes opérationnels au sein de votre organisation, certains widgets vous permettent de regrouper les données en fonction du compte, de la région et de la balise.
- Clés de balise de génération de rapports : Lorsque vous configurez Explorer , vous pouvez spécifier jusqu'à cinq clés de balise. Ces clés vous aident à regrouper et filtrer les données dans Explorer. Si une clé spécifiée correspond à une clé sur une ressource qui génère un OpsItem , la clé et la valeur sont incluses dans le OpsItems .
- Trois modes d'affichage des comptes et régions AWS : Explorer inclut les modes d'affichage suivants pour OpsData et OpsItems dans les comptes et régions AWS :
  1. Un seul compte/une seule région : Il s'agit de la vue par défaut. Ce mode permet aux utilisateurs d'afficher les données et OpsItems à partir de leur propre compte et de la région actuelle.
  2. Un seul compte/plusieurs régions : Ce mode nécessite que vous créiez une ou plusieurs synchronisations de données de ressource à l'aide de la Explorer page Settings (Paramètres). Une synchronisation des données de ressource est agrégée OpsData à partir d'une ou plusieurs régions. Après avoir créé une synchronisation des données de ressource, vous pouvez changer de synchronisation à utiliser dans le tableau de bord Explorer. Vous pouvez ensuite filtrer et regrouper les données en fonction de la région.
  3. Plusieurs comptes/plusieurs régions : Ce mode nécessite que votre organisation ou votre entreprise utilise AWS Organizations toutes les fonctions activées. Une fois que vous avez configuré AWS Organizations dans votre environnement informatique, vous pouvez regrouper toutes les données de compte dans un compte principal . Vous pouvez ensuite créer des synchronisations de données de ressource afin de filtrer et de regrouper les données en fonction de la région. Pour de plus amples informations sur le mode Toutes les fonctions d'Organisations, consultez Activation de toutes les fonctions de votre organisation.
- Génération de rapports : Vous pouvez exporter Explorer les rapports sous forme de fichiers .csv séparés par des virgules vers un compartiment Amazon Simple Storage Service (Amazon S3). Vous recevez une alerte de la part d'Amazon Simple Notification Service (Amazon SNS) lorsqu'une exportation est terminée.

Quel est le lien entre Explorer et OpsCenter ?

Systems ManagerOpsCenter (p. 176)fournit un emplacement central où les ingénieurs d'exploitation et les professionnels de l'informatique voient, étudient et résolvent OpsItems les ressources AWS. Explorer est un hub de rapports où DevOps les responsables affichent des résumés agrégés de leurs données d'opérations, y compris OpsItems , entre les régions et les comptes AWS. Explorer aide les utilisateurs à découvrir des tendances et des tendances, et, si nécessaire, à résoudre rapidement les problèmes à l'aide des runbooks Systems Manager Automation.

OpsCenter est maintenant intégré au programme d'installation d'Explorer. Si vous avez déjà configuré OpsCenter, Explorer présente automatiquement les données opérationnelles, y compris les informations agrégées concernant les OpsItems. Si vous n'avez pas configuré OpsCenter, vous pouvez utiliser le programme d'installation d'Explorer pour commencer avec les deux fonctions. Pour plus d'informations, consultez la section Démarrer avec Systems Manager Explorer et OpsCenter (p. 157).

Présentation du OpsData

OpsDatacorrespond à toutes les données d'opérations affichées dans le tableau de bord Systems Manager de l'explorateur. Explorer récupère OpsData à partir des sources suivantes :

- Amazon Elastic Compute Cloud (Amazon EC2)

  Les données affichées dans Explorer sont notamment les suivantes : nombre total d'instances, nombre total d'instances gérées et non gérées, et nombre d'instances qui utilisent une image AMI (Amazon Machine Image) spécifique.

- Systems Manager OpsCenter

  Les données affichées dans Explorer sont notamment les suivantes : un nombre d'OpsItems par statut, un nombre d'OpsItems par gravité, un nombre d'OpsItems ouverts entre groupes et sur des périodes de 30 jours, et des données historiques d'OpsItems au fil du temps.
- Systems Manager Gestionnaire de correctifs

  Les données affichées dans Explorer incluent un nombre d'instances qui ne sont pas compatibles avec les correctifs.
- AWS Trusted Advisor

  Les données affichées dans Explorer incluent : l'état des vérifications des bonnes pratiques pour les instances réservées EC2 dans les domaines de l'optimisation des coûts, de la sécurité, de la tolérance aux pannes, des performances et des limites de service.
- AWS Compute Optimizer

  Les données affichées dans Explorer comprennent : le nombre d'instances EC2 sous provisionnées et surdimensionnées, les résultats d'optimisation, les détails de tarification à la demande et les recommandations pour le type d'instance et le prix.
- Dossiers du Centre AWS Support

  Les données affichées dans Explorer incluent : l'ID de cas, la gravité, l'état, l'heure de création, l'objet, le service et la catégorie.

  > **Note**
  >
  > Pour afficher AWS Trusted Advisor les cas et le Centre AWS Support dans Explorer , vous devez avoir un compte d'entreprise ou un compte Business configuré avec AWS Support.

Vous pouvez afficher et gérer les OpsData sources à partir de la page Explorer Settings (Paramètres de l'Explorateur). Pour plus d'informations sur l'installation et la configuration des services qui remplissent les Explorer widgets avec OpsData , consultez Configuration des services connexes (p. 157) .

L'utilisation d'Explorer entraîne-t-elle des frais ?

Oui. Lorsque vous activez les règles par défaut pour la création d'OpsItems au cours de l'installation intégrée, vous lancez un processus qui crée automatiquement des OpsItems. Votre compte est facturé en fonction du nombre d'OpsItems créés par mois. Votre compte est également facturé en fonction du nombre d'appels d'API `GetOpsItem`, `DescribeOpsItem`, `UpdateOpsItem` et `GetOpsSummary` effectués par mois. En outre, vous pouvez vous voir facturer les appels d'API publics vers d'autres services qui présentent des informations de diagnostic pertinentes. Pour de plus amples informations, veuillez consulter Tarification AWS Systems Manager.

Rubriques

- Démarrer avec Systems Manager Explorer et OpsCenter (p. 157)
- Utilisation de Systems Manager Explorer (p. 168)
- Exportation OpsData à partir deSystems ManagerExplorer (p. 172)
- Résolution des problèmes liés à Systems Manager Explorer (p. 174)

Démarrer avec Systems Manager Explorer et OpsCenter

Systems Manager utilise une expérience d'installation intégrée pour vous aider à mettre en route Systems Manager Explorer et Systems Manager OpsCenter. Dans cette documentation, le programme d'installation d'Explorer et OpsCenter est désigné par le mot Installation intégrée. Si vous avez déjà configuré OpsCenter, vous devez tout de même terminer l'installation intégrée pour vérifier les paramètres et les options. Si vous n'avez pas configuré OpsCenter, vous pouvez utiliser l'installation intégrée pour commencer avec les deux fonctions. L'installation intégrée effectue les tâches suivantes.

1. Configure les rôles et les autorisations (p. 159) : L'installation intégrée crée un rôle AWS Identity and Access Management (IAM) qui permet Amazon EventBridge à de créer automatiquement OpsItems en fonction des règles par défaut. Après la configuration, vous devez configurer les autorisations d'utilisateur, de groupe ou de rôle IAM pour OpsCenter, comme décrit dans cette section.
2. Active les règles par défaut pour OpsItem la création (p. 162) : L'installation intégrée crée des règles par défaut dans EventBridge . Ces règles créent automatiquement des OpsItems en réponse à des événements. Voici des exemples de ces événements : changement d'état pour une ressource AWS, modification des paramètres de sécurité ou indisponibilité d'un service.
3. Active les OpsData sources (p. 163) : L'installation intégrée active les sources de données qui renseignent les Explorer widgets.
4. Permet de spécifier des clés de balise de génération de rapports (p. 163) : L'installation intégrée vous permet de spécifier jusqu'à cinq clés de balise de génération de rapports à affecter automatiquement aux nouveaux OpsItems qui répondent à des critères spécifiques.

Une fois que vous avez terminé l'installation intégrée, nous vous recommandons de configurer Explorer pour afficher les données de plusieurs régions et comptes (p. 163), Explorer et OpsCenter synchroniser OpsData automatiquement OpsItems et pour le compte et la région AWS que vous avez utilisés lorsque vous avez terminé l'installation intégrée. Vous pouvez regrouper OpsData et OpsItems à partir d'autres comptes et régions en créant une synchronisation des données de ressource.

> **Note**
>
> Vous pouvez modifier les configurations d'installation à tout moment sur la page Paramètres.

Configuration des services connexes

Explorer et OpsCenter collectent des informations auprès d'autres services AWS et fonctionnalités de Systems Manager. Ils interagir également avec eux. Nous vous recommandons d'installer et de configurer ces autres services ou fonctionnalités avant d'utiliser l'installation intégrée.

Le tableau suivant répertorie les tâches qui permettent à Explorer et OpsCenter de collecter des informations auprès d'autres services AWS et fonctionnalités de Systems Manager et d'interagir avec eux.

Tâche	Informations
Vérifier les autorisations dans Systems Manager Automation	Explorer et OpsCenter permettent de corriger les problèmes concernant les ressources AWS en utilisant des documents Systems Manager Automation (runbooks). Pour utiliser cette fonctionnalité de correction, vous devez disposer d'une autorisation pour exécuter des documents d'automatisation Systems Manager. Pour plus d'informations, consultez la section Configuration d'Automation (p. 308).

Tâche	Informations
Installer et configurerSystems ManagerGestionnaire de correctifs	Explorer comprend un widget qui fournit des informations sur la conformité des correctifs. Pour afficher ces données dans Explorer, vous devez configurer l'application de correctifs. Pour plus d'informations, consultez la section AWS Systems Manager Gestionnaire de correctifs (p. 1047).
Installer et configurerSystems ManagerState Manager	Explorercomprend un widget qui fournit des informations sur la conformité des State Manager associations. Pour afficher ces données dans Explorer , vous devez configurer State Manager . Pour plus d'informations, consultez la section AWS Systems Manager State Manager (p. 987).
Activer l'Enregistreur de configuration AWS Config	Explorer utilise les données fournies par l'enregistreur de configuration AWS Config pour renseigner les widgets avec des informations relatives à vos instances EC2. Pour afficher ces données dans Explorer, activez l'enregistreur de configuration AWS Config. Pour de plus amples informations, veuillez consulter Gestion de l'enregistreur de configuration. Note Après avoir activé l'enregistreur de configuration, Systems Manager peut prendre jusqu'à six heures pour afficher les données dans les widgets Explorer qui présentent des informations sur vos instances EC2.
Activer AWS Trusted Advisor	Explorer utilise les données fournies par Trusted Advisor pour afficher l'état des vérifications des bonnes pratiques pour les instances réservées Amazon EC2 dans les domaines de l'optimisation des coûts, de la sécurité, de la tolérance aux pannes, des performances et des limites de service. Pour afficher ces données dans Explorer, vous devez disposer d'un plan de support commercial ou d'entreprise. Pour de plus amples informations, consultez AWS Support.
Activer AWS Compute Optimizer	Explorer utilise les données fournies par Compute Optimizer pour afficher les détails du nombre d'instances EC2 sous provisionnées et sur provisionnées, les résultats d'optimisation, les détails de tarification à la demande et des recommandations pour le type d'instance et le prix. Pour afficher ces données dans Explorer, activez Compute Optimizer. Pour de plus amples informations, veuillez consulter Démarrez avec AWS Compute Optimizer.

Configuration des rôles et des autorisations pour Systems Manager Explorer

L'installation intégrée crée et configure automatiquement les rôles IAM pour Systems Manager Explorer et OpsCenter. Si vous avez terminé l'installation intégrée, vous n'avez pas besoin d'effectuer de tâches supplémentaires pour configurer les rôles et les autorisations pour Explorer. Toutefois, vous devez configurer l'autorisation pour OpsCenter, comme décrit ultérieurement dans cette rubrique.

Table des matières

- À propos des rôles créés par l'installation intégrée (p. 159)
- Configuration d'autorisations pour Systems Manager OpsCenter (p. 160)

À propos des rôles créés par l'installation intégrée

L'installation intégrée crée et configure les rôles suivants pour travailler avec Explorer et OpsCenter.

- AWSServiceRoleForAmazonSSM: Fournit l'accès aux ressources AWS gérées ou utilisées par Systems Manager .
- OpsItem-CWE-Role : Active CloudWatch Events et EventBridge pour créer OpsItems en réponse à des événements courants.
- AWSServiceRoleForAmazonSSMDécouverte de compte _ : Permet Systems Manager à d'appeler d'autres services AWS pour découvrir les informations de compte AWS lors de la synchronisation des données. Pour de plus amples informations sur ce rôle, consultez À propos du rôle AWSServiceRoleForAmazonSSM _AccountDiscovery (p. 159).
- AmazonSSMExplorerExport: Permet Explorer d'exporter OpsData vers un fichier CSV (valeurs séparées par des virgules).

À propos du rôle AWSServiceRoleForAmazonSSM _AccountDiscovery

Si vous configurez Explorer pour afficher les données provenant de plusieurs comptes et régions à l'aide d'AWS Organizations et d'une synchronisation des données de ressource, Systems Manager crée un rôle lié au service. Systems Manager utilise ce rôle pour obtenir des informations sur vos comptes AWS dans AWS Organizations. Le rôle utilise la stratégie d'autorisations suivante.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "organizations:DescribeAccount",
            "organizations:DescribeOrganization",
            "organizations:ListAccounts",
            "organizations:ListAWSServiceAccessForOrganization",
            "organizations:ListChildren",
            "organizations:ListParents"
         ],
         "Resource":"*"
      }
   ]
}
```

Pour plus d'informations sur le rôle AWSServiceRoleForAmazonSSM _AccountDiscovery, consultez Utilisation des rôles pour collecter des informations de compte AWS pour Systems Manager Explorer (p. 1315) .

Configuration d'autorisations pour Systems Manager OpsCenter

Après avoir terminé l'installation intégrée, vous devez configurer les autorisations d'utilisateur, de groupe ou de rôle IAM afin que les utilisateurs puissent effectuer des actions dans OpsCenter.

Avant de commencer

OpsItems peut être consulté ou modifié uniquement dans le compte où il a été créé. Vous ne pouvez pas partager ou transférer OpsItems entre plusieurs comptes AWS. Pour cette raison, nous vous recommandons de configurer les autorisations OpsCenter dans le compte AWS qui est utilisé pour exécuter vos charges de travail AWS. Vous pouvez ensuite créer des utilisateurs ou groupes IAM dans ce compte. De cette manière, les ingénieurs de plusieurs opérations ou professionnels de l'informatique peuvent créer, afficher et modifier OpsItems dans le même compte AWS.

Explorer et OpsCenter utilisent les actions d'API suivantes. Vous pouvez utiliser toutes les fonctions d'Explorer et d'OpsCenter si votre utilisateur, votre groupe ou votre rôle IAM a accès à ces actions. Vous pouvez également créer un accès plus restrictif, comme décrit plus loin dans cette section.

- CreateOpsItem
- CreateResourceDataSync
- DescribeOpsItems
- DeleteResourceDataSync
- GetOpsItem
- GetOpsSummary
- ListResourceDataSync
- UpdateOpsItem
- UpdateResourceDataSync

La procédure suivante décrit comment ajouter une stratégie en ligne d'accès complet à un utilisateur IAM. Si vous préférez, vous pouvez spécifier l'autorisation de lecture seule en affectant la stratégie en ligne suivante à un compte d'utilisateur, groupe ou rôle.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ssm:GetOpsItem",
        "ssm:GetOpsSummary",
        "ssm:DescribeOpsItems"
      ],
      "Resource": "*"
    }
  ]
}
```

Pour de plus amples informations sur la création et la modification de stratégies IAM, veuillez consulter Création de stratégies IAM dans le IAM Guide de l'utilisateur. Pour de plus amples informations sur la façon d'attribuer cette stratégie à un groue IAM, veuillez consulter Attacher une stratégie à un groupe IAM.

1. Connectez-vous à la AWS Management Console et ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation, sélectionnez Users (Utilisateurs).
3. Dans la liste, choisissez un nom.

4. Choisissez l'onglet Autorisations.
5. Sur le côté droit de la page, sous Permission policies (Stratégies d'autorisation), choisissez Add inline policy (Ajouter une stratégie en ligne).
6. Choisissez l'onglet JSON.
7. Remplacez le contenu par défaut par ce qui suit :

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ssm:GetOpsItem",
        "ssm:UpdateOpsItem",
        "ssm:DescribeOpsItems",
        "ssm:CreateOpsItem",
        "ssm:CreateResourceDataSync",
        "ssm:DeleteResourceDataSync",
        "ssm:ListResourceDataSync",
        "ssm:UpdateResourceDataSync"

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

8. Choisissez Examiner une stratégie.
9. Sur la page Examiner une stratégie, dans le champ Nom, saisissez un nom pour la stratégie en ligne. Exemple : **`OpsCenter-Access-Full`**.
10. Choisissez Create policy (Créer une stratégie).

Restriction de l'accès aux OpsItems à l'aide de balises

Vous pouvez également limiter l'accès de OpsItems à l'aide d'une stratégie IAM en ligne qui spécifie des balises. La stratégie utilise le format suivant.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "One_or_more_OpsItem_API_actions"
             ],
      "Resource": "*"
   ,
   "Condition": { "StringEquals": { "ssm:resourceTag/tag_key": "tag_value" } }
    }
  ]
}
```

Voici un exemple qui spécifie une clé de balise Service et une valeur de balise Finance. Avec cette stratégie, l'utilisateur peut uniquement appeler l'action GetOpsItem d'API pour afficher OpsItems les éléments qui ont été précédemment balisés avec Key=Department et Value=Finance. Les utilisateurs ne peuvent pas afficher les autres OpsItems.

```
{
```

```
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ssm:GetOpsItem"
             ],
      "Resource": "*"
   ,
   "Condition": { "StringEquals": { "ssm:resourceTag/Department": "Finance" } }
    }
  ]
}
```

Voici un exemple qui spécifie les actions d'API pour l'affichage et la mise à jour d'OpsItems. Cette stratégie spécifie également deux ensembles de paires clé-valeur de balise : Département des Finances et Projet Unity.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "ssm:GetOpsItem",
            "ssm:UpdateOpsItem"
         ],
         "Resource":"*",
         "Condition":{
            "StringEquals":{
               "ssm:resourceTag/Department":"Finance",
               "ssm:resourceTag/Project":"Unity"
            }
         }
      }
   ]
}
```

Pour plus d'informations sur l'ajout de balises à un OpsItem, consultez Création manuelle de OpsItems (p. 189).

Activation des règles par défaut

L'installation intégrée configure automatiquement les règles par défaut suivantes dans EventBridge. Ces règles créent des OpsItems dans Systems Manager OpsCenter. Si vous ne souhaitez pas que EventBridge crée des OpsItems pour les événements suivants, désactivez cette option dans l'installation intégrée. Si vous préférez, vous pouvez spécifier OpsCenter comme cible d'événements EventBridge spécifiques. Pour plus d'informations, consultez la section Configuration de EventBridge pour créer automatiquement des OpsItems pour les événements spécifiques (p. 188). Vous pouvez également désactiver les règles par défaut à tout moment sur la page Paramètres .

> **Important**
>
> Actuellement, vous ne pouvez pas modifier les valeurs Catégorie et Gravité pour les règles par défaut, mais vous pouvez modifier ces valeurs sur les OpsItems règles créées à partir des règles par défaut.

Rule	Category	Severity
CWE rules (11)		
SSMOpsItems-Autoscaling-instance-launch-failure	Availability	2-High
SSMOpsItems-Autoscaling-instance-termination-failure	Availability	2-High
SSMOpsItems-EBS-snapshot-copy-failed	Availability	2-High
SSMOpsItems-EBS-snapshot-creation-failed	Availability	2-High
SSMOpsItems-EBS-volume-performance-issue	Performance	3-Medium
SSMOpsItems-EC2-issue	Availability	2-High
SSMOpsItems-EC2-scheduled-change	Availability	3-Medium
SSMOpsItems-RDS-issue	Availability	2-High
SSMOpsItems-RDS-scheduled-change	Availability	3-Medium
SSMOpsItems-SSM-maintenance-window-execution-failed	Availability	3-Medium
SSMOpsItems-SSM-maintenance-window-execution-timedout	Availability	2-High

Configuration des OpsData sources

L'installation intégrée active les sources de données suivantes qui remplissent les widgets de l'explorateur de Systems Manager.

- Amazon Elastic Compute Cloud (Amazon EC2)
- Systems Manager OpsCenter
- Conformité Systems Manager Gestionnaire de correctifs des correctifs
- Conformité Systems Manager State Manager des associations
- AWS Trusted Advisor
- AWS Compute Optimizer

Spécification des clés de balise

Lorsque vous configurez Systems Manager Explorer, vous pouvez spécifier jusqu'à cinq clés de balise de génération de rapports. Ces clés de balise doivent déjà exister sur vos ressources AWS. Ce ne sont pas de nouvelles clés de balise. Après avoir ajouté les clés au système, vous pouvez filtrer OpsItems dans Explorer utilisant ces clés de balise.

> **Note**
>
> Vous pouvez également spécifier des clés de balise de génération de rapports sur la page Paramètres.

Configuration de Systems Manager Explorer de sorte à afficher les données de plusieurs comptes et régions

Systems Manager utilise une expérience d'installation intégrée pour vous aider à mettre en route Systems Manager Explorer etSystems Manager OpsCenter. Après avoir terminé l'installation intégrée, Explorer et OpsCenter synchronisent automatiquement les données. Plus spécifiquement, ces fonctionnalités se synchronisent OpsData et OpsItems pour le compte et la région AWS que vous avez utilisés lorsque vous avez terminé l'installation intégrée. Pour regrouper OpsData et à OpsItems partir d'autres comptes

et régions, vous devez créer une synchronisation des données de ressource, comme décrit dans cette rubrique.

> **Note**
>
> Pour plus d'informations sur la configuration intégrée, veuillez consulter Démarrer avec Systems Manager Explorer et OpsCenter (p. 157).

A propos de la synchronisation des données des ressources pour Explorer

La synchronisation des données des ressources pour Explorer propose deux options d'agrégation :

- Un seul compte/plusieurs régions : Vous pouvez configurer Explorer pour agréger OpsItems les OpsData données de plusieurs régions AWS, mais l'ensemble de données est limité au compte AWS actuel.
- Plusieurs comptes/plusieurs régions : Vous pouvez configurer Explorer pour regrouper les données de plusieurs régions et comptes AWS. Cette option demande la configuration d'AWS Organizations. Une fois que vous avez installé et configuré AWS Organizations , vous pouvez regrouper les données dans Explorer par unité d'organisation ou pour l'ensemble d'une organisation. Systems Manager regroupe les données dans l' AWS Organizations compte principal avant de les afficher dans Explorer . Pour de plus amples informations, veuillez consulter Présentation d'AWS Organizations dans le Manuel de l'utilisateur AWS Organizations.

Le schéma suivant montre une synchronisation des données des ressources configurée pour fonctionner avec AWS Organizations. Dans ce scénario, l'utilisateur a deux comptes définis dans AWS Organizations. La synchronisation des données de ressource regroupe les données des deux comptes et de plusieurs régions AWS dans l' AWS Organizations compte principal où elles sont ensuite affichées dans Explorer .

Avant de commencer

Prenez note des informations importantes suivantes concernant la synchronisation des données de ressource pour Explorer.

- Explorer prend en charge un maximum de cinq synchronisations de données de ressources.
- Vous devez effectuer l'installation intégrée dans chaque compte et région où vous souhaitez que l'Explorer accède aux données. Dans le cas contraire, n'affiche pas Explorer et OpsData pour ces comptes et régions dans lesquels vous n'avez pas terminé l'installation intégrée.OpsItems
- Pour afficher OpsData et OpsItems afficher plusieurs comptes , le AWS Organizations mode Toutes les fonctions doit être activé et vous devez être connecté à .AWS Organizationscompte principal
- Après avoir créé une synchronisation des données de ressource pour une région, vous ne pouvez pas modifier les options de compte pour cette synchronisation. Par exemple, si vous créez une synchronisation dans la région us-east-2 (Ohio) et que vous choisissez l'option Include only the current account (Inclure uniquement le compte courant) vous ne pouvez pas modifier cette synchronisation ultérieurement et choisir l'option Include all accounts from my AWS Organizations configuration (Inclure tous les comptes de ma configuration d'AWS Organizations). Vous devez supprimer la première synchronisation des données de ressource et en créer une nouvelle. Pour plus d'informations, veuillez consulter Suppression d'une synchronisation des données de ressource Systems ManagerExplorer (p. 172)
- est OpsData en lecture seule.Explorer

Utilisez la procédure suivante pour créer une synchronisation de données de ressource pour Explorer.

Pour créer une synchronisation de données de ressources

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Explorer.
3. Sélectionnez Paramètres.
4. Dans la section Configure resource data sync (Configurer la synchronisation des données de ressource), choisissez Create resource data sync (Créer une synchronisation des données de ressource).
5. Pour Resource data sync name (Nom de synchronisation des données de ressource), entrez un nom.
6. Dans la section Add accounts (Ajouter des comptes) choisissez une option.

   Note

   Pour utiliser l'une ou l'autre des AWS Organizations options, vous devez être connecté à ou vous devez être connecté à un compte administrateur délégué.AWS Organizationscompte principalExplorer Pour de plus amples informations sur le compte d'administrateur délégué, veuillez consulter Configuration d'un administrateur délégué (p. 166).
7. Dans la section Regions to include (Régions à inclure) choisissez l'une des options suivantes.
   - Choisissez All current and future regions (Toutes les régions actuelles et futures) pour synchroniser automatiquement les données de toutes les régions AWS actuelles et de toutes les nouvelles régions qui seront en ligne à l'avenir.
   - Choisissez All regions (Toutes les régions) pour synchroniser automatiquement les données de toutes les régions AWS actuelles.
   - Choisissez individuellement les régions à inclure.
8. Choisissez Create resource data sync (Créer une synchronisation des données de ressource).

Le système peut prendre plusieurs minutes pour renseigner Explorer avec les données après la création d'une synchronisation des données de ressource. Vous pouvez afficher la synchronisation en la

sélectionnant dans la liste Select a resource data sync (Sélectionner une synchronisation des données de ressource) dans Explorer.

Configuration d'un administrateur délégué

Si vous agrégez des données Explorerprovenant de plusieurs régions et comptes AWS à l'aide de la synchronisation des données de ressources avec AWS Organizations, nous vous suggérons de configurer un administrateur délégué pour Explorer. Un administrateur délégué améliore la sécurité Explorer de différentes façons.

- Vous limitez le nombre d'administrateurs Explorer qui peuvent créer ou supprimer des synchronisations de données de ressources de plusieurs comptes et régions à une seule personne.
- Vous n'avez plus besoin d'être connecté à AWS Organizations compte principal pour administrer la synchronisation des données de ressource dans Explorer .

Pour de plus amples informations sur la synchronisation des données de ressource, veuillez consulter Configuration de Systems Manager Explorer de sorte à afficher les données de plusieurs comptes et régions (p. 163). Pour de plus amples informations sur AWS Organizations, veuillez consulter Qu'est-ce qu'AWS Organizations ? dans le Manuel de l'utilisateur AWS Organizations.

Rubriques

- Avant de commencer (p. 166)
- Configurer un administrateur délégué Explorer (p. 166)
- Désenregistrer un administrateur délégué Explorer (p. 167)

Avant de commencer

La liste suivante contient des informations importantes sur l'administration déléguée Explorer.

- Vous ne pouvez déléguer qu'un seul compte pour l'administration Explorer.
- L'ID de compte que vous spécifiez en tant qu'administrateur délégué Explorerdoit être répertorié en tant que compte membre dans AWS Organizations. Pour de plus amples informations, consultez Création d'un compte AWS dans votre organisation dans le Manuel de l'utilisateur AWS Organizations.
- Un administrateur délégué peut utiliser toutes les actions de l'API de synchronisation des données de ressources Explorer dans la console ou à l'aide d'outils de programmation tels que le SDK, l'AWS CLI, ou Outils AWS pour Windows PowerShell. Les actions de l'API de synchronisation des données de ressource sont les suivantes : CreateResourceDataSync, DeleteResourceDataSync , et ListResourceDataSync .UpdateResourceDataSync
- Un administrateur délégué peut rechercher, filtrer et agréger des données Explorer dans la console ou à l'aide d'outils de programmation tels que le SDK, l'AWS CLI ou Outils AWS pour Windows PowerShell. Les fonctions de recherche, de filtre et d'agrégation de données utilisent l'action GetOpsSummary d'API .
- Les synchronisations de données de ressource créées par un administrateur délégué ne sont disponibles que dans le compte d'administrateur délégué. Vous ne pouvez pas afficher les synchronisations ou les données agrégées dans l' AWS Organizations compte principal .
- Un administrateur délégué peut créer un maximum de cinq synchronisations de données de ressource.
- Un administrateur délégué peut créer une synchronisation des données de ressource pour une organisation entière dans AWS Organizations ou un sous-ensemble d'unités d'organisation.

Configurer un administrateur délégué Explorer

Pour enregistrer un administrateur délégué Explorer, procédez comme suit.

Pour enregistrer un administrateur délégué Explorer

1. Connectez-vous à votre AWS Organizations compte principal .
2. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
3. Dans le panneau de navigation, choisissez Explorer.
4. Sélectionnez Paramètres.
5. Dans la section Administrateur délégué pour Explorer, vérifiez que vous avez configuré les options de rôle lié au service et d'accès au service requis. Si nécessaire, choisissez les boutons pour configurer ces options.

6. Dans Account ID (ID de compte), entrez l'ID de compte AWS. Ce compte doit être un compte membre dans AWS Organizations.
7. Choisissez Enregistrer l'administrateur délégué.

L'administrateur délégué a désormais accès aux options AWS Organizations suivantes de la page Créer une synchronisation des données de ressource.

Désenregistrer un administrateur délégué Explorer

Pour annuler l'inscription d'un administrateur délégué Explorer, procédez comme suit. L'annulation de l'inscription d'un compte administrateur délégué ne peut être effectuée que par l' AWS Organizations compte principal . Lorsque l'inscription d'un compte d'administrateur délégué est annulé, le système

supprime toutes les synchronisations de données de ressource AWS Organizations créées par l'administrateur délégué.

Pour annuler l'inscription d'un administrateur délégué Explorer

1. Connectez-vous à votre AWS Organizations compte principal .
2. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
3. Dans le panneau de navigation, choisissez Explorer.
4. Sélectionnez Paramètres.
5. Dans la section Administrateur délégué pour Explorer, choisissez Désenregistrer. Le système affiche un avertissement.
6. Tapez l'ID du compte et choisissez Supprimer.

Le compte n'a plus accès aux actions de l'API de synchronisation des données de ressource AWS Organizations. Le système supprime toutes les synchronisations de données de ressource AWS Organizations créées par le compte.

Utilisation de Systems Manager Explorer

Cette section contient des informations sur la façon de personnaliser Systems Manager Explorer en modifiant la disposition du widget et les données affichées dans le tableau de bord.

Table des matières

- Modification des règles par défaut pour OpsItems (p. 168)
- Modification des sources de données Systems Manager Explorer (p. 169)
- Personnalisation de l'affichage et utilisation de filtres (p. 170)
- Suppression d'une synchronisation des données de ressource Systems ManagerExplorer (p. 172)

Modification des règles par défaut pour OpsItems

Lorsque vous avez terminé l'installation intégrée, le système active plus d'une douzaine de règles dans Amazon EventBridge. Ces règles créent automatiquement des OpsItems dans Systems Manager OpsCenter. Systems Manager Explorer affiche ensuite des informations agrégées concernant les OpsItems.

Chaque règle comprend des valeurs Catégorie et Gravité prédéfinies. Lorsque le système crée des OpsItems à partir d'un événement, il affecte automatiquement la Catégorie et la Gravitéprédéfinies.

> **Important**
>
> Actuellement, vous ne pouvez pas modifier les valeurs Catégorie et Gravité pour les règles par défaut, mais vous pouvez modifier ces valeurs sur les OpsItems règles créées à partir des règles par défaut.

Rule	Category	Severity
CWE rules (11)		
SSMOpsItems-Autoscaling-instance-launch-failure	Availability	2-High
SSMOpsItems-Autoscaling-instance-termination-failure	Availability	2-High
SSMOpsItems-EBS-snapshot-copy-failed	Availability	2-High
SSMOpsItems-EBS-snapshot-creation-failed	Availability	2-High
SSMOpsItems-EBS-volume-performance-issue	Performance	3-Medium
SSMOpsItems-EC2-issue	Availability	2-High
SSMOpsItems-EC2-scheduled-change	Availability	3-Medium
SSMOpsItems-RDS-issue	Availability	2-High
SSMOpsItems-RDS-scheduled-change	Availability	3-Medium
SSMOpsItems-SSM-maintenance-window-execution-failed	Availability	3-Medium
SSMOpsItems-SSM-maintenance-window-execution-timedout	Availability	2-High

Pour modifier les règles par défaut de création d’OpsItems

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Explorer.
3. Sélectionnez Paramètres.
4. Dans la sectionRègles d'OpsItems, choisissez Modifier.
5. Développez les CWE rules (Règles CWE).
6. Désactivez la case à cocher en regard des règles que vous ne souhaitez pas utiliser.
7. Utilisez les listes Catégorie et Gravité pour modifier ces informations pour une règle.
8. Choisissez Save.

Vos modifications prennent effet à la prochaine création par le système crée d'un OpsItem.

Modification des sources de données Systems Manager Explorer

Systems Manager Explorer affiche les données provenant des sources suivantes.

- Amazon Elastic Compute Cloud (Amazon EC2)
- Systems Manager OpsCenter
- Conformité Systems Manager Gestionnaire de correctifs des correctifs
- Conformité Systems Manager State Manager des associations
- AWS Trusted Advisor
- AWS Compute Optimizer
- Dossiers du Centre AWS Support

> **Note**
>
> - Pour afficher les cas du Centre AWS Support dans Explorer , vous devez avoir un compte Entreprise ou Business configuré avec AWS Support.

- Vous ne pouvez pas configurer Explorer de sorte à cesser d'afficher des données OpsCenter OpsItem.

Avant de commencer

Vérifiez que vous configurez les services qui remplissent les widgets Explorer avec des données. Pour plus d'informations, consultez la section Configuration des services connexes (p. 157).

Pour modifier des sources de données

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Explorer.
3. Sélectionnez Paramètres.
4. Dans la section OpsData Sources, choisissez Edit (Modifier).
5. Développez OpsData les sources.
6. Ajouter ou supprimer une ou plusieurs sources.
7. Choisissez Save.

Personnalisation de l'affichage et utilisation de filtres

Vous pouvez personnaliser la disposition des widgets dans Systems Manager Explorer grâce à la fonction de glisser-déposer. Vous pouvez également personnaliser l' OpsData et l' OpsItems affiché dans à l'Exploreraide de filtres, comme décrit dans cette rubrique.

Personnalisation de la disposition des widgets

Utilisez la procédure suivante pour personnaliser la disposition des widgets dans Explorer.

Pour personnaliser la disposition des widgets

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Explorer.
3. Choisissez un widget à déplacer.
4. Cliquez longuement sur le nom du widget, puis faites-le glisser vers son nouvel emplacement.

5. Répétez ce processus pour chaque widget que vous souhaitez repositionner.

Si vous décidez que vous n'aimez pas la nouvelle disposition, choisissez Reset layout (Réinitialiser la disposition) pour déplacer tous les widgets à leur emplacement d'origine.

Utilisation de filtres pour modifier les données affichées dans Explorer

Par défaut, Explorer affiche les données du compte AWS actuel et de la région actuelle. Si vous créez une ou plusieurs synchronisations de données de ressource, vous pouvez utiliser des filtres pour changer la synchronisation active. Vous pouvez ensuite choisir d'afficher les données pour une région spécifique ou pour toutes les régions. Vous pouvez également utiliser la barre de recherche pour filtrer selon différents critères OpsItem et key-tag.

Pour modifier les données affichées dans Explorer à l'aide de filtres

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Explorer.
3. Dans la section Filtre, utilisez la liste Select a resource data sync (Sélectionner une synchronisation des données de ressource) pour choisir une synchronisation.
4. Utilisez la liste Régions pour choisir une région AWS spécifique ou choisissez All regions (Toutes les régions).
5. Choisissez la barre de recherche, puis choisissez les critères selon lesquels filtrer les données.

6. Appuyez sur Entrée.

Explorer conserve les options de filtre sélectionnées si vous fermez et rouvrez la page.

Suppression d'une synchronisation des données de ressource Systems ManagerExplorer

Dans Systems Manager Explorer , vous pouvez regrouper OpsData et OpsItems à partir d'autres comptes et régions en créant une synchronisation des données de ressource.

Vous ne pouvez pas modifier les options de compte pour une synchronisation de données de ressource. Par exemple, si vous avez créé une synchronisation dans la région us-east-2 (Ohio) et que vous avez choisi l'option Include only the current account (Inclure uniquement le compte courant), vous ne pouvez pas modifier cette synchronisation ultérieurement et choisir l'option Include all accounts from my AWS Organizations configuration (Inclure tous les comptes de ma configuration d'AWS Organizations). Vous devez supprimer la synchronisation des données de ressource et en créer une nouvelle, comme décrit dans la procédure suivante.

Pour supprimer une synchronisation de données de ressources

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Explorer.
3. Sélectionnez Paramètres.
4. Dans la section Configure resource data sync (Configurer la synchronisation des données de ressource), choisissez la synchronisation des données de ressource que vous souhaitez supprimer.
5. Sélectionnez Supprimer.

Exportation OpsData à partir deSystems ManagerExplorer

Lorsque vous cliquez sur un lien dans Systems Manager Explorer , certaines pages s'affichent OpsData dans une liste. Ces pages incluent un bouton Export qui vous permet d'exporter jusqu'à 5 000 OpsData éléments sous la forme d'un fichier .csv (valeurs séparées par une virgule) vers un compartiment Amazon Simple Storage Service (Amazon S3). Pour pouvoir exporter des données à partir de pages Explorer, vous devez configurer l'exportation de données comme décrit dans cette rubrique.

Avant de commencer

Lorsque vous configurez l'exportation de données, vous devez spécifier une rubrique Amazon Simple Notification Service (Amazon SNS) qui existe dans la même région AWS que celle où vous souhaitez exporter les données. Systems Manager envoie une notification à la rubrique Amazon SNS lorsqu'une exportation est terminée. Pour plus d'informations sur la création d'une Amazon SNS rubrique , consultez Didacticiel : Création d'une Amazon SNS rubrique .

De plus, sachez que lorsque vous exportez Explorer des données, Systems Manager crée un rôle AWS Identity and Access Management (IAM) nommé `AmazonSSMExplorerExportRole` . Ce rôle utilise la IAM stratégie suivante.

```
{
      "Version": "2012-10-17",
      "Statement": [
          {
              "Effect": "Allow",
              "Action": [
                  "s3:PutObject"
              ],
              "Resource": [
                 "arn:aws:s3:::{{ExportDestinationS3BucketName}}/*"
              ]
          },
          {
              "Effect": "Allow",
              "Action": [
                  "s3:GetBucketAcl"
              ],
              "Resource": [
                 "arn:aws:s3:::{{ExportDestinationS3BucketName}}"
              ]
          },
          {
              "Effect": "Allow",
              "Action": [
                  "sns:Publish"
              ],
              "Resource": [
                  "{{SnsTopicArn}}"
              ]
          },
          {
              "Effect": "Allow",
              "Action": [
                  "logs:DescribeLogGroups",
                  "logs:DescribeLogStreams"
              ],
              "Resource": [
                  "*"
              ]
          },
          {
              "Effect": "Allow",
              "Action": [
                  "logs:CreateLogGroup",
                  "logs:PutLogEvents",
                  "logs:CreateLogStream"
              ],
              "Resource": [
                  "*"
              ]
          },
          {
```

```
                "Effect": "Allow",
                "Action": [
                    "ssm:GetOpsSummary"
                ],
                "Resource": [
                    "*"
                ]
            }
        ]
    }
```

Le rôle comprend l'entité d'approbation suivante.

```
{
      "Version": "2012-10-17",
      "Statement": [
        {
          "Effect": "Allow",
          "Principal": {
            "Service": "ssm.amazonaws.com"
          },
          "Action": "sts:AssumeRole"
        }
      ]
    }
```

Pour exporter OpsData à partir deExplorer

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Explorer.
3. Sélectionnez Paramètres.
4. Dans la section Configure data export (Configurer l'exportation de données) choisissez Edit (Modifier).
5. Pour charger le fichier d'exportation de données vers un compartiment existant, choisissez Select an existing S3 bucket (Sélectionner un compartiment S3 existant), puis choisissez le compartiment dans la liste. Pour charger le fichier d'exportation de données vers un nouveau compartiment , choisissez Create a new S3 bucket (Créer un nouveau compartiment S3), puis entrez le nom que vous souhaitez utiliser pour le nouveau compartiment.
6. Utilisez la liste Select an Amazon SNS topic ARN (Sélectionner un ARN de rubrique Amazon SNS) pour choisir la rubrique à notifier une fois l'exportation terminée.
7. Sélectionnez Create.

Vous pouvez désormais exporter OpsData des Explorer pages vers le compartiment S3 spécifié.

Si vous ne pouvez pas exporter des données à l'aide de cette procédure, vérifiez que votre compte IAM utilisateur, votre groupe ou votre rôle inclut les `iam:CreatePolicyVersion` actions `iam:DeletePolicyVersion` et . Pour plus d'informations sur l'ajout de ces actions à votre compte utilisateur, groupe ou rôle, consultez Modification des stratégies IAM dans le IAM Guide de l'utilisateur .

Résolution des problèmes liés à Systems Manager Explorer

Cette rubrique contient des informations sur la façon de résoudre les problèmes courants avec Systems Manager Explorer.

Les AWS Organizations options de la page Create resource data sync (Créer une synchronisation des données de ressource) sont grisées

Les options suivantes de la page Créer une synchronisation de données de ressource ne sont disponibles que si vous avez installé et configuré AWS Organizations. Si vous avez installé et configuré AWS Organizations , l' ou un administrateur AWS Organizations délégué peut créer des synchronisations de données de ressource qui utilisent ces options.compte principalExplorer

Pour de plus amples informations, veuillez consulter Configuration de Systems Manager Explorer de sorte à afficher les données de plusieurs comptes et régions (p. 163) et Configuration d'un administrateur délégué (p. 166).

Explorer n'affiche aucune donnée du tout

- Vérifiez que vous avez terminé l'installation intégrée dans chaque compte et région où vous souhaitez qu'Explorer accède aux données et les affiche. Dans le cas contraire, n'affiche pas Explorer et OpsData pour ces comptes et régions dans lesquels vous n'avez pas terminé l'installation intégrée.OpsItems Pour plus d'informations, consultez la section Démarrer avec Systems Manager Explorer et OpsCenter (p. 157).
- Lorsque vous utilisez Explorer pour afficher des données de plusieurs comptes et régions, vérifiez que vous êtes connecté à .AWS Organizationscompte principal Pour afficher OpsData et OpsItems afficher plusieurs comptes et régions, vous devez être connecté à ce compte.

Les widgets concernant des instances EC2 n'affichent pas de données

Si les widgets concernant les instances EC2, par exemple Instance count (Nombre d'instances), Managed instances (Instances gérées) et Instance by AMI (Instances par AMI) n'affichent pas de données, vérifiez les points suivants :

- Assurez-vous d'avoir attendu quelques minutes. OpsDatapeut prendre plusieurs minutes pour s'afficher dans Explorer une fois que vous avez terminé l'installation intégrée.
- Vérifiez que vous avez configuré l'enregistreur de configuration AWS Config. Explorer utilise les données fournies par l'enregistreur de configuration AWS Config pour renseigner les widgets avec des informations relatives à vos instances EC2. Pour de plus amples informations, veuillez consulter Gestion de l'enregistreur de configuration.
- Vérifiez que la Amazon EC2 OpsData source est activée sur la page Settings (Paramètres). Vérifiez également que plus de 6 heures se sont écoulées depuis que vous avez activé l'enregistreur de configuration ou depuis que vous avez apporté des modifications à vos instances. Systems Manager peut prendre jusqu'à six heures pour afficher les données d'AWS Config dans les widgets Explorer EC2 après la première activation de l'enregistreur de configuration ou si vous apportez des modifications à vos instances.
- Sachez que si une instance est arrêtée ou interrompue, Explorer cesse d'afficher ces instances après 24 heures.
- Vérifiez que vous êtes dans la région AWS dans laquelle vous avez configuré vos instances EC2. Explorer n'affiche pas de données concernant les instances sur site.
- Si vous avez configuré une synchronisation des données de ressource pour plusieurs comptes et régions, vérifiez que vous êtes connecté à Organisations compte principal .

Le widget d'application de correctifs n'affiche pas les données

Le widget Non-compliant instances for patching (Instances non conformes pour l'application de correctifs) affiche uniquement les données relatives aux instances de correctifs non conformes. Ce widget n'affiche aucune donnée si vos instances sont conformes. Si vous pensez avoir des instances non conformes, vérifiez que vous avez installé et configuré Systems Manager l'application de correctifs et utilisez Gestionnaire de correctifs pour vérifier la conformité de vos correctifs. Pour plus d'informations, consultez la section AWS Systems Manager Gestionnaire de correctifs (p. 1047).

Questions diverses

Explorer ne permet pas de modifier ou de corriger les OpsItems : lesOpsItems affichées dans les comptes ou régions sont en lecture seule. Ils ne peuvent être mis à jour et corrigés qu'à partir de leur compte ou de leur région d'origine.

AWS Systems Manager OpsCenter

OpsCenter offre un emplacement centralisé où les ingénieurs d'exploitation et les professionnels de l'informatique peuvent consulter, étudier et résoudre des éléments de travail opérationnels (OpsItems) associés aux ressources AWS. OpsCenter est conçu pour réduire la durée moyenne de résolution des problèmes ayant un impact sur les ressources AWS. Cette fonctionnalité Systems Manager regroupe et normalise les OpsItems entre les services tout en fournissant des données d'investigation contextuelles sur chaque OpsItem, sur les OpsItems associés et sur les ressources connexes. OpsCenter fournit également des documents Automation Systems Manager (runbooks) que vous pouvez utiliser pour résoudre rapidement les problèmes. Vous pouvez spécifier des données personnalisées consultables pour chaque OpsItem. Vous pouvez également afficher des rapports récapitulatifs sur les OpsItems générés automatiquement, par statut et par source.

OpsCenter est intégré à Amazon EventBridge. Cela signifie que vous pouvez créer des règles EventBridge qui créent automatiquement des éléments OpsItems pour n'importe quel service AWS qui publie des événements sur EventBridge. Par exemple, vous pouvez configurer SSM OpsItems comme cible pour les types d'événements suivants, et des centaines d'autres :

- Les questions de sécurité, telles que les alertes de AWS Security Hub
- Les problèmes de performance, comme un événement de limitation pour Amazon DynamoDB ou les performances de volume de Amazon Elastic Block Store dégradées (EBS)
- Les défaillances, tel qu'un échec de groupe de Amazon EC2 Auto Scaling pour lancer une instance ou un échec d'exécution de l'automatisation de Systems Manager
- Les alertes d'intégrité, telles qu'une alerte AWS Health pour la maintenance planifiée
- Les changements d'état, tels qu'un changement de l'état de l'instance EC2 passant de `Running` à `Stopped`

OpsCenter est également intégré à Amazon CloudWatch Application Insights pour .NET et SQL Server. Cela signifie que vous pouvez créer automatiquement des éléments OpsItems pour les problèmes détectés dans vos applications.

Les ingénieurs d'exploitation et les professionnels de l'informatique peuvent créer, afficher et modifier des éléments OpsItems en utilisant la page OpsCenter de la console AWS Systems Manager, des actions d'API publiques, le AWS CLI, Outils AWS pour Windows PowerShell ou les kits de développement logiciel (SDK) AWS. Vous pouvez également utiliser AWS Lambda avec Amazon SNS pour créer des éléments OpsItems à partir de sources telles que les alarmes CloudWatch. Les actions d'API publiques OpsCenter vous permettent également d'intégrer OpsCenter à vos systèmes de gestion des cas aux tableaux de bord d'intégrité.

Comment mon organisation peut-elle tirer parti de OpsCenter ?

AWS Systems Manager OpsCenter permet une expérience standard et unifiée pour l'affichage, l'utilisation et la correction des problèmes liés aux ressources AWS. Une expérience unifiée et standard améliore le temps nécessaire pour corriger les problèmes, étudier les problèmes liés à la sécurité et former de nouveaux ingénieurs d'exploitation et professionnels de l'informatique. Une expérience standard et unifiée réduit aussi le nombre d'erreurs manuelles introduites dans le système lors de la gestion et de la correction des problèmes.

Plus spécifiquement, OpsCenter offre les avantages suivants pour les ingénieurs d'exploitation et les organisations :

- Vous n'avez plus besoin de parcourir plusieurs pages de la console pour afficher, étudier et résoudre des éléments OpsItems liés aux ressources AWS. Les éléments OpsItems sont regroupés pour tous les services dans un emplacement central.
- Vous pouvez afficher des données spécifiques au service et contextuelles pour les OpsItems qui sont automatiquement générés par Amazon EventBridge et CloudWatch Application Insights pour .NET et SQL Server.
- Vous pouvez spécifier l'Amazon Resource Name (ARN) d'une ressource liée à un OpsItem. En spécifiant les ressources connexes, OpsCenter utilise la logique intégrée pour éviter la création de OpsItems en double.
- Vous pouvez afficher des détails et des informations de résolution pour des OpsItems similaires.
- Vous pouvez rapidement afficher des informations sur et exécuter Systems Manager Documents d’automatisation (runbooks) pour résoudre les problèmes.

Quelles sont les fonctions d'OpsCenter ?

- Automatisé et manuel OpsItem création

  OpsCenter est intégré à Amazon EventBridge. Cela signifie que vous pouvez créer des règles EventBridge qui créent automatiquement des éléments OpsItems pour n'importe quel service AWS qui publie des événements sur EventBridge. Vous pouvez également créer manuellement des éléments OpsItems.

  OpsCenter est également intégré à Amazon CloudWatch Application Insights pour .NET et SQL Server. Cela signifie que vous pouvez créer automatiquement des éléments OpsItems pour les problèmes détectés dans vos applications.
- Détaillé et consultable OpsItems

  Chaque OpsItem comprend plusieurs champs d'informations, y compris le titre, l'ID, la priorité, la description, la source de OpsItem, ainsi que la date/heure de sa dernière mise à jour. Chaque OpsItem inclut également les fonctionnalités configurables suivantes :
  - État: Ouvert, En cours, Résolu ou Ouvert et En cours.
  - Ressources connexes: Une ressource connexe est la ressource affectée ou la ressource qui a déclenché l'événement Amazon EventBridge qui a créé le OpsItem. Chaque OpsItem comprend une section Ressources connexes où OpsCenter répertorie automatiquement l'ARN (Amazon Resource Name) de la ressource connexe. Vous pouvez également spécifier manuellement les ARN de ressources connexes. Pour certains types d'ARN, OpsCenter crée automatiquement un lien ciblé qui permet d'afficher des détails sur la ressource sans avoir à consulter d'autres pages de la console pour afficher ces informations. Par exemple, si vous spécifiez l'ARN d'une instance EC2, vous pouvez afficher toutes les détails fournis à EC2 sur cette instance dans OpsCenter. Vous pouvez ajouter manuellement les ARN des ressources connexes supplémentaires. Chaque OpsItem peut répertorier

un maximum de 100 ARN de ressources connexes. Pour plus d'informations, consultez la section Utilisation des ressources connexes (p. 193).

- Liées et similaires OpsItems: Le Lié OpsItems vous permet de spécifier les ID de OpsItems qui sont d'une certaine manière liées à la OpsItem. La fonctionnalité OpsItem similaire examine automatiquement des titres et des descriptions OpsItem, puis répertorie d'autres OpsItems qui peuvent être liés ou qui vous intéressent.
- Données opérationnelles consultables et privées: Les données opérationnelles sont des données personnalisées qui fournissent des détails sur le OpsItem. Par exemple, vous pouvez spécifier les fichiers journaux, les chaînes d'erreur, les clés de licence, les conseils de dépannage ou d'autres données pertinentes. Vous saisissez les données opérationnelles sous forme de paires clé-valeur. La clé possède une longueur maximale de 128 caractères. La valeur a une taille maximale de 20 Ko.

  Ces données personnalisées pouvant faire l'objet de recherches, mais avec des restrictions. Pour la fonctionnalité Données opérationnelles interrogeables, tous les utilisateurs ayant accès à la page Présentation d'OpsItem (telle que fournie par l'action d'API Describe OpsItem) peuvent afficher et faire des recherches sur les données spécifiées. Pour la fonctionnalité Données opérationnelles privées, les données sont uniquement visibles par les utilisateurs qui ont accès au OpsItem (tel que fourni par l'action d'API GetOpsItem).
- Déduplication: En spécifiant les ressources connexes, OpsCenter utilise la logique intégrée pour éviter la création de OpsItems en double. De plus, les OpsItems qui sont automatiquement créées à partir d'un événement dans EventBridge incluent une chaîne de déduplication afin de réduire le nombre de OpsItems dupliqués. Pour plus d'informations, consultez la section Réduction de OpsItems en double (p. 198).

- Résolution simplifiée à l'aide de runbooks

  Chaque OpsItem comprend une section Runbooks avec une liste des documents d'automatisation Systems Manager que vous pouvez utiliser pour corriger automatiquement les problèmes courants avec les ressources AWS. Si vous ouvrez un OpsItem, choisissez une ressource AWS pour ce OpsItem, puis choisissez le Exécuter l'automatisation dans la console, puis OpsCenter fournit une liste des documents Automation que vous pouvez exécuter sur la ressource AWS qui a généré le OpsItem. Après avoir exécuté un document Automation (runbook) à partir d'un OpsItem, le runbook est automatiquement associé à la ressource associée du OpsItem pour référence ultérieure. De plus, si vous configurez automatiquement OpsItem règles dans Amazon EventBridge en utilisant OpsCenter, puis EventBridge associe automatiquement les runbooks pour les événements courants. OpsCenter conserve un enregistrement de 30 jours des documents Automation exécutés pour un OpsItem. Pour plus d'informations, consultez la section Correction des problèmes OpsItem avec Systems Manager Automation (p. 200).
- Notification de modification: Vous pouvez spécifier l'ARN d'un Amazon Simple Notification Service (SNS) et publiez des notifications à tout moment OpsItem est modifié(e) ou modifié(e). La rubrique SNS doit exister dans la même région AWS que le composant OpsItem.
- Capacités de recherche OpsItem étendues : OpsCenter fournit plusieurs options de recherche pour vous aider à trouver rapidement OpsItems. Voici quelques exemples de recherche : OpsItem ID, Titre, Heure de dernière modification, Valeur des données opérationnelles, Source et ID d'automatisation d'une exécution de runbook, pour n'en citer que quelques-uns. Vous pouvez également limiter les résultats de recherche en utilisant le statut des filtres.
- OpsItem rapports récapitulatifs

  OpsCenter comprend une page de rapport récapitulatif qui affiche automatiquement les sections suivantes :

  - Résumé de statut : un résumé de OpsItems par statut (Ouvert, En cours, Résolu, Ouvert et En cours).
  - Sources avec la plupart des OpsItems ouverts : une répartition des principaux services AWS à des OpsItems ouverts.
  - OpsItems par source et âge : nombre d'OpsItems regroupées par source et jours depuis la création.

Pour plus d'informations sur l'affichage des rapports récapitulatifs de OpsCenter, consultez Affichage des rapports de synthèse OpsCenter (p. 206).

- IAM contrôle d'accès

  En utilisant AWS Identity and Access Management (IAM), vous pouvez contrôler quels membres de votre organisation peuvent créer, afficher, répertorier et mettre à jour OpsItems. Vous pouvez également affecter des balises à OpsItems, puis créer des stratégies IAM qui donnent accès à des utilisateurs et des groupes en fonction des balises. Pour plus d'informations, consultez la section Démarrez avec OpsCenter (p. 181).

- Prise en charge de la capacité de journalisation et d'audit

  Vous pouvez contrôler et consigner les actions des utilisateurs de OpsCenter dans votre compte AWS via l'intégration avec d'autres services AWS. Pour plus d'informations, consultez la section Audit et journalisation de l'activité de OpsCenter (p. 210).

- Accès à la console, à l'interface de ligne de commande, à PowerShell et au kit SDK OpsCenter des capacités

  Vous pouvez utiliser OpsCenter via la console AWS Systems Manager, l'AWS CLI, les Outils AWS pour PowerShell ou le kit SDK AWS de votre choix.

Comment OpsCenter fonctionne-t-il avec Amazon EventBridge ? Quel service dois-je utiliser ?

Amazon EventBridge fournit un flux d'événements système en temps quasi réel qui décrivent les modifications apportées aux ressources AWS. A l'aide de règles simples et rapidement configurées, vous pouvez faire correspondre des événements et les acheminer vers un ou plusieurs flux ou une ou plusieurs fonctions cibles. En règle générale, EventBridge vous permet de savoir qu'il y a un problème avec vos ressources.

OpsCenter vous aide à étudier et corriger le problème. OpsCenter rassemble des données EventBridge ou des données saisies manuellement par des ingénieurs, afin qu'ils puissent effectuer un examen approfondi. OpsCenter fournit également des runbooks Automation pour corriger rapidement ces problèmes. OpsCenter s'intègre à EventBridge en vous permettant de créer automatiquement OpsItems (ou vous pouvez créer manuellement OpsItems) afin de prendre en compte les types de problèmes suivants : dégradation des performances, changements d'état, défaillances d'exécution, maintenance des notifications et alertes de sécurité.

OpsCenter s'intègre-t-il à mon système actuel de gestion des cas ?

OpsCenter est conçu pour compléter vos systèmes de gestion des cas existant. Vous pouvez intégrer OpsItems à votre système de gestion de cas à l'aide des actions d'API publiques. Vous pouvez également conserver votre flux de travail du cycle de vie manuelle dans les systèmes actuels et utiliser OpsCenter comme hub d'enquête et de résolution.

Pour plus d'informations sur les actions d'API publiques OpsCenter, consultez les actions d'API suivantes dans le Référence d'API AWS Systems Manager.

- CreateOpsItem
- DescribeOpsItems
- GetOpsItem

- GetOpsSummary
- UpdateOpsItem

L'utilisation d'OpsCenter entraîne-t-elle des frais ?

Oui. Pour de plus amples informations, veuillez consulter Tarification AWS Systems Manager.

Est-ce que OpsCenter fonctionne avec mes instances gérées locales et hybrides ?

Oui. Vous pouvez utiliser OpsCenter pour examiner et corriger les problèmes liés à vos instances gérées sur site qui sont configurés pour Systems Manager. Pour de plus amples informations sur l'installation et la configuration des serveurs et des machines virtuelles pour Systems Manager, veuillez consulter Configuration d'AWS Systems Manager pour les environnements hybrides (p. 49).

Quelles sont les limites des ressources pour OpsCenter ?

Resource	Limite par défaut
Nombre total de OpsItems autorisé par compte et par région AWS (notamment Ouvertet Résolu OpsItems)	500 000
Nombre maximal de règles OpsItems par compte et par mois	10 000
Taille maximale de la valeur de données opérationnelles	20 Ko
Nombre maximal de runbooks Automation associés par OpsItem	10
Nombre maximal d'exécutions d'automatisation de runbooks stockées dans des données opérationnelles sous un seul runbook associé	10
Nombre maximal de ressources connexes que vous pouvez spécifier par OpsItem	100
Nombre maximal d'OpsItems connexes que vous pouvez spécifier par OpsItem	10
Longueur maximale d'une chaîne de déduplication	64 caractères
Durée avant un OpsItem est automatiquement archivé par le système (quel que soit le statut)	36 mois

Rubriques

- Démarrez avec OpsCenter (p. 181)
- Création de OpsItems (p. 188)

- Utilisation des OpsItems (p. 193)
- Correction des problèmes OpsItem avec Systems Manager Automation (p. 200)
- Affichage des rapports de synthèse OpsCenter (p. 206)
- Référence des ressources prises en charge (p. 206)
- Audit et journalisation de l'activité de OpsCenter (p. 210)

Démarrez avec OpsCenter

Le programme d'installation de Systems Manager OpsCenter est intégré à la configuration pour Systems Manager Explorer. Explorer est un tableau de bord des opérations personnalisable qui présente des informations sur vos ressources AWS. Explorer affiche une vue agrégée des données d'opérations (OpsData) pour vos comptes AWS et pour toutes les régions. Dans Explorer, les données opérationnelles englobent les métadonnées relatives à vos instances EC2, les détails de conformité des correctifs et les éléments de travail opérationnels (OpsItems). Explorer fournit un contexte sur la répartition des OpsItems entre vos unités commerciales ou applications, leur évolution au fil du temps et leur variation par catégorie. Vous pouvez regrouper et filtrer les informations dans Explorer pour vous concentrer sur les éléments qui vous intéressent et qui nécessitent une action. Lorsque vous identifiez des problèmes prioritaires, vous pouvez utiliser Systems Manager OpsCenter pour exécuter des runbooks Automation et résoudre rapidement ces problèmes.

Si vous avez déjà configuré OpsCenter, vous devez tout de même terminer l'installation intégrée pour vérifier les paramètres et les options. Si vous n'avez pas configuré OpsCenter, vous pouvez utiliser l'installation intégrée pour commencer avec les deux fonctions. Pour plus d'informations, consultez la section Démarrer avec Systems Manager Explorer et OpsCenter (p. 157).

(Facultatif) Recevoir les notifications OpsItem

Vous pouvez configurer OpsCenter pour envoyer des notifications à une rubrique Amazon Simple Notification Service (Amazon SNS) lorsque les systèmes créent une nouvelle OpsItem ou mettent à jour une rubrique OpsItem existante. Effectuez les tâches suivantes pour recevoir des notifications pour OpsItems.

- Tâche 1 : Créer une rubrique Amazon SNS et s'y abonner (p. 181)
- Tâche 2 : Mettre à jour le Amazon SNS stratégie d’accès (p. 182)
- Tâche 3 : Mettre à jour le AWS KMS stratégie d’accès (facultatif) (p. 183)
- Tâche 4 : Activer par défaut OpsItems règles pour envoyer des notifications pour les nouveaux OpsItems (p. 184)

Tâche 1 : Créer une rubrique Amazon SNS et s'y abonner

Pour recevoir des notifications, vous devez créer et vous abonner à une rubrique Amazon SNS. Pour de plus amples informations, veuillez consulter Création d'une rubrique et Abonnement d'un point de terminaison à une rubrique Amazon SNS dans le Amazon Simple Notification Service Manuel du développeur.

> **Note**
>
> Pour recevoir des notifications, vous devez spécifier le nom de ressource Amazon (ARN) d'une rubrique Amazon SNS qui se trouve dans la même région et le même compte AWS que l’OpsItem. Si vous utilisez OpsCenter dans plusieurs régions ou comptes, vous devez créer et vous abonner à une rubrique Amazon SNS dans chaque région ou compte où vous souhaitez recevoir les notifications OpsItem.

Tâche 2 : Mettre à jour le Amazon SNS stratégie d’accès

Utilisez la procédure suivante pour mettre à jour la stratégie d'accès Amazon SNS afin que Systems Manager puisse publier les notifications OpsItem dans la rubrique Amazon SNS créée dans la tâche 1.

1. Connectez-vous à la AWS Management Console et ouvrez la console Amazon SNS à l’adresse https://console.aws.amazon.com/sns/v3/home.
2. Dans le panneau de navigation, sélectionnez Rubriques.
3. Choisissez la rubrique que vous avez créée dans la tâche 1, puis choisissez Edit (Modifier).
4. Développez la stratégie d'accès.

5. Ajoutez le bloc Sid suivant à la stratégie existante.

```
{
      "Sid": "Allow OpsCenter to publish to this topic",
      "Effect": "Allow",
      "Principal": {
        "Service": "ssm.amazonaws.com"
      },
      "Action": "SNS:Publish",
       "Resource": "arn:aws:sns:AWS_Region:account_ID:topic_name"
    }
```

Entrez ce bloc après le bloc Sid existant. Dans l'exemple suivant, le nouveau bloc est entré à la ligne 29.

6. Sélectionnez Save Changes (Enregistrer les modifications).

Le système envoie désormais les notifications à la rubrique Amazon SNS quand des OpsItems sont créés ou mis à jour.

> **Important**
>
> Si vous avez configuré la rubrique Amazon SNS avec une clé de chiffrement côté serveur AWS Key Management Service (AWS KMS), vous devez terminer la tâche 3. Si vous souhaitez configurer les OpsItems créés par les règles OpsItem par défaut pour publier dans la rubrique Amazon SNS, vous devez également terminer la tâche 4.

Tâche 3 : Mettre à jour le AWS KMS stratégie d'accès (facultatif)

Si vous avez activé le chiffrement côté serveur AWS Key Management Service (AWS KMS) pour votre rubrique Amazon SNS, vous devez également mettre à jour la stratégie d'accès de la clé CMK AWS KMS que vous avez choisie lors de la configuration de la rubrique. Utilisez la procédure suivante pour mettre à jour la stratégie d'accès afin que Systems Manager puisse publier les notifications OpsItem dans la rubrique Amazon SNS créée dans la tâche 1.

> **Note**
>
> OpsCenter ne prend pas en charge la publication des OpsItems sur une rubrique Amazon SNS configurée avec une clé gérée AWS.

1. Ouvrez la console AWS KMS à l'adresse https://console.aws.amazon.com/kms.
2. Pour changer de région AWS, utilisez le sélecteur de région dans l'angle supérieur droit de la page.
3. Dans le volet de navigation, choisissez Customer managed keys (Clés gérées par le client).
4. Choisissez l'ID de la clé CMK que vous avez choisie lors de la création de la rubrique.
5. Dans la section Key policy (Stratégie de clé), choisissez Switch to policy view (Passer à la vue de stratégie).

6. Choisissez Edit.
7. Ajoutez le bloc Sid suivant à la stratégie existante.

```
{
      "Sid": "Allow OpsItems to decrypt the key",
      "Effect": "Allow",
      "Principal": {
        "Service": "ssm.amazonaws.com"
      },
      "Action": ["kms:Decrypt", "kms:GenerateDataKey*"],
       "Resource": "arn:aws:kms:AWS_Region:account_ID:key/key_ID"
    }
```

Entrez ce bloc après l'un des blocs Sid existants. Dans l'exemple suivant, le nouveau bloc est entré à la ligne 14.

8. Sélectionnez Save Changes (Enregistrer les modifications).

Tâche 4 : Activer par défaut OpsItems règles pour envoyer des notifications pour les nouveaux OpsItems

Les règles OpsItems par défaut dans Amazon EventBridge ne sont pas configurées avec un ARN pour les notifications Amazon SNS. Suivez la procédure suivante pour modifier une règle dans EventBridge et entrer un bloc `notifications`.

Pour ajouter un bloc de notifications à une règle OpsItem par défaut

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez OpsCenter.
3. Choisissez l'onglet OpsItems, puis choisissez Configurer les sources.

4. Choisissez la règle source que vous souhaitez configurer avec un bloc `notification`, comme illustré dans l'exemple suivant :

5. Sur la page des détails de la règle, choisissez Edit (Modifier).

6. Faites défiler jusqu'à la section Select targets (Sélectionner les cibles).

7. Entrez le bloc `notifications` avant le bloc `resources`, comme indiqué ici.

8. Faites défiler jusqu'au bas de la page des détails de la règle et choisissez Update (Mettre à jour).

La prochaine fois que le système crée un OpsItem pour la règle par défaut, il publie une notification dans la rubrique Amazon SNS.

(Facultatif) Créer des instructions OpsItem pour votre organisation

Nous recommandons que chaque organisation crée un ensemble d'instructions simples qui favorisent la cohérence lors de la création et de la modification de OpsItems. Instructions permettre aux utilisateurs de localiser et de résoudre rapidement OpsItems. Les recommandations pour votre organisation doivent définir les bonnes pratiques lorsque les utilisateurs entrent des informations dans les champs OpsItem suivants.

> **Note**
>
> Amazon EventBridge remplit les champs Titre, Source et Description du OpsItems généré automatiquement. Vous pouvez modifier les champs Titre et Description, mais vous ne pouvez pas modifier le champ Source.

Field	Description
Titre	Consignes doivent encourager une expérience de OpsItem dénomination constant. Par exemple,

Field	Description
	vos instructions peuvent nécessiter que chaque titre comprenne des informations sur la ressource affectée, le statut, l'environnement et le nom ou l'alias de l'ingénieur qui travaille activement sur le problème, le cas échéant. Tous les OpsItems créés par EventBridge incluent un titre qui décrit l'événement ayant entraîné la création de OpsItem, mais vous pouvez modifier ces titres. Vous pouvez rechercher dans OpsItems Titre :contient. Si vos instructions d'attribution de noms encouragent l'utilisation cohérente de mots-clés, vous améliorez vos résultats de recherche.
Source	Les instructions peuvent inclure des ID spécifiant les logiciels, les numéros de version (le cas échéant) ou d'autres données pertinentes pour aider les utilisateurs à identifier l'origine du problème. Vous ne pouvez pas modifier le champ Source après la création de OpsItem.
Priority	(Facultatif) Les instructions incluent la définition de la priorité la plus élevée et la plus faible pour votre organisation et tout accord de niveau de service en fonction de la priorité. Vous pouvez spécifier la priorité de 1 à 5.
Description	Les instructions doivent proposer le nombre de détails sur le problème à inclure et toutes les étapes (le cas échéant) permettant de reproduire le problème.
Notifications	Les instructions doivent indiquer quel Amazon Resource Name (ARN) de rubrique Amazon Simple Notification Service (SNS) spécifier lors de la création ou de la modification de OpsItems. Sachez que les notifications SNS sont spécifiques à une région. Cela signifie que vous devez spécifier un ARN qui se trouve dans la même région AWS que le composant OpsItem.
Ressources connexes	Les instructions peuvent inclure des détails sur les ressources qui doivent ou ne doivent pas avoir d'ARN spécifié. Pour les types de ressources AWS pris en charge, l'ARN crée un lien ciblé pour plus de détails sur la ressource.

Field	Description
Données opérationnelles	Vous pouvez spécifier des données personnalisées pour chaque OpsItem qui apportent des indications sur le problème et d'autres données pertinentes pour référence ultérieure. Vous pouvez spécifier des données personnalisées interrogeables. Tous les utilisateurs ayant accès à la page Présentation de OpsItem peuvent rechercher et afficher ces données. Vous pouvez également spécifier des données personnalisées privées qui sont uniquement visibles par les utilisateurs ayant accès à ce OpsItem. Les instructions peuvent spécifier la structure et les normes de paires clé-valeur. Ces paires clé-valeur peuvent décrire les données opérationnelles et les détails de résolution, afin d'améliorer les résultats de recherche.

Création de OpsItems

Lorsque vous configurez pour la première fois OpsCenter à l'aide de l'installation intégrée, vous activez Amazon EventBridge pour qu'il crée automatiquement des OpsItems conformément à des règles communes. Les procédures de cette section décrivent comment configurer un événement ou une alarme spécifique pour créer OpsItems. Pour de plus amples informations sur l'activation des règles EventBridge par défaut pour la création d'OpsItems à l'aide de l'installation intégrée, veuillez consulter Démarrer avec Systems Manager Explorer et OpsCenter (p. 157).

Table des matières

- Configuration de EventBridge pour créer automatiquement des OpsItems pour les événements spécifiques (p. 188)
- Configuration de CloudWatch Application Insights pour.NET et SQL Server à créer automatiquement OpsItems (p. 189)
- Création manuelle de OpsItems (p. 189)

Configuration de EventBridge pour créer automatiquement des OpsItems pour les événements spécifiques

Utilisez la procédure suivante pour configurer Éléments d’opérations SSM en tant que cible d’un EventBridge événement. Quand EventBridge reçoit l'événement, il crée un nouveau OpsItem dans OpsCenter.

Pour configurer la fonctionnalité OpsCenter en tant que cible d'un événement CloudWatch

1. Ouvrez la console Amazon EventBridge à l'adresse https://console.aws.amazon.com/events/.
2. Dans le volet de navigation, choisissez Règle, puis choisissez de créer une nouvelle règle ou de modifier une règle existante.
3. Après avoir vérifié les détails de la règle, pour Sélectionner un événement bus, choisissez Bus d’événement AWS par défaut.
4. Pour Cible , choisissez Élément d’opérations SSM.
5. Développer Configurer l’entrée et vérifiez que Événement correspondant est sélectionné.

6. Choisir Créer un nouveau rôle pour cette ressource spécifique pour créer un nouveau rôle avec les autorisations requises. Ou choisissez Use existing role (Utiliser un rôle existant), puis choisissez le rôle IAM que vous avez créée qui donne à CloudWatch l'autorisation de créer OpsItems dans OpsCenter. Pour plus d'informations sur les autorisations et les rôles requis, consultez Démarrez avec OpsCenter (p. 181).
7. Si vous créez une nouvelle règle, choisissez Créer.

   -ou-

   Si vous modifiez une règle, choisissez Mettre à jour.
8. Dans le volet de navigation, choisissez Rules. Si vous avez créé une nouvelle règle, vérifiez que la liste inclut une nouvelle règle qui commence par `AwsSSMOpsCenter-*`comme `AwsSSMOpsCenter-EC2` ou `AwsSSMOpsCenter-DynamoDB`.

Une fois OpsItem créé à partir d'un événement, vous pouvez afficher les détails de l'événement en ouvrant OpsItem et en faisant défiler jusqu'à la section Données opérationnelles privées.

Pour plus d'informations sur la configuration des options dans un OpsItem, consultez Utilisation des OpsItems (p. 193).

Configuration de CloudWatch Application Insights pour.NET et SQL Server à créer automatiquement OpsItems

OpsCenter est intégré à Amazon CloudWatch Application Insights pour .NET et SQL Server. Cela signifie que vous pouvez créer automatiquement des éléments OpsItems pour les problèmes détectés dans vos applications. Pour de plus amples informations sur la configuration d'Application Insights pour créer des OpsItems, veuillez consulter Configuration de votre application dans le Guide de l'utilisateur Amazon CloudWatch.

Création manuelle de OpsItems

Cette section inclut des procédures pour créer manuellement OpsItems pour les problèmes qui ne sont pas créés automatiquement par Amazon EventBridge.

Avant de commencer

Si vous créez manuellement un OpsItem pour la ressource AWS impactée, collectez des informations sur cette ressource pour que vous puissiez créer un Amazon Resource Name (ARN). Si vous spécifiez un ARN lorsque vous créez un ensemble de données OpsItem, OpsCenter crée automatiquement un lien ciblé pour obtenir des informations détaillées sur la ressource. Par exemple, si vous spécifiez l'ARN pour une instance EC2 impactée, OpsCenter crée un lien ciblé vers les détails relatifs à cette instance. Pour plus d'informations sur la création d'un ARN, consultez Amazon Resource Names (ARN) et espaces de noms AWS Service dans le Référence générale d'Amazon Web Services.

> **Note**
>
> OpsCenter ne prend pas en charge la création de liens ciblés pour tous les types d'ARN. Pour afficher une liste des ressources qui prennent en charge les liens ciblés basés sur les ARN, consultez Référence des ressources prises en charge (p. 206).

Cette section comprend les procédures suivantes.

- Pour créer manuellement un OpsItem (console) (p. 190)
- Pour créer manuellement un OpsItem (AWS CLI) (p. 190)

Pour créer manuellement un OpsItem (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez OpsCenter.
3. Sélectionnez CreateOpsItem. Si vous ne voyez pas ce bouton, cliquez sur l'onglet OpsItems, puis choisissez Créer des OpsItem.
4. Dans Titre, saisissez un nom descriptif pour vous aider à comprendre l'objectif de l'OpsItem.
5. Pour Source, saisissez le type de ressource AWS impacté ou d'autres informations source pour aider les utilisateurs à comprendre l'origine de l'OpsItem.

   Note

   Vous ne pouvez pas modifier le champ Source après avoir créé OpsItem.
6. Pour Priorité, choisissez le niveau de priorité.
7. Pour Description, saisissez des informations à propos de cet OpsItem y compris (le cas échéant) les étapes de reproduction du problème.
8. Pour Deduplication string (Chaîne de déduplication), saisissez les mots que le système doit utiliser pour rechercher les OpsItems dupliqués. Pour plus d'informations sur les chaînes de déduplication, consultez Réduction de OpsItems en double (p. 198).
9. (Facultatif) Pour Notifications, spécifiez l'ARN de la rubrique SNS dans laquelle vous souhaitez envoyer des notifications lorsque OpsItem est mis à jour. Vous devez spécifier un ARN Amazon SNS qui se trouve dans la même région AWS que le composant OpsItem.
10. (Facultatif) Sous Ressources connexes, choisissez Ajouter pour spécifier l'ARN de la ressource impactée et toute ressource connexe.
11. Sélectionnez CreateOpsItem.

En cas de réussite, la OpsItem s'ouvre. Pour plus d'informations sur la configuration des options dans un OpsItem, consultez Utilisation des OpsItems (p. 193).

Pour créer manuellement un OpsItem (AWS CLI)

1. Ouvrez la AWS CLI et exécutez la commande suivante pour créer un OpsItem.

```
aws ssm create-ops-item --title "Descriptive_title" --description
 "Information_about_the_issue" --priority Number_between_1_and_5 --
source Source_of_the_issue --operational-data Up_to_20_KB_of_data_or_path_to_JSON_file
 --notifications Arn="SNS_ARN_in_same_Region" --tags "Key=key_name,Value=a_value"
```

   Voici quelques exemples d'.

   Linux

```
aws ssm create-ops-item --title "EC2 instance disk full" --description "Log
 clean up may have failed which caused the disk to be full" --priority 2 --source
 ec2 --operational-data '{"EC2":{"Value":"12345","Type":"SearchableString"}}'
 --notifications Arn="arn:aws:sns:us-west-1:12345678:TestUser1" --tags
 "Key=EC2,Value=ProductionServers"
```

   La commande suivante utilise la clé `/aws/resources` dans OperationalData pour créer un OpsItem avec une ressource Amazon DynamoDB connexe.

```
aws ssm create-ops-item --title "EC2 instance disk full" --description "Log clean
 up may have failed which caused the disk to be full" --priority 2 --source ec2
 --operational-data '{"/aws/resources":{"Value":"[{\"arn\": \"arn:aws:dynamodb:us-
west-2:12345678:table/OpsItems\"}]","Type":"SearchableString"}}' --notifications
 Arn="arn:aws:sns:us-west-2:12345678:TestUser"
```

La commande suivante utilise la clé `/aws/automations` dans OperationalData pour créer un OpsItem qui spécifie le document AWS-ASGEnterStandby en tant que runbook Automation associé.

```
aws ssm create-ops-item --title "EC2 instance disk full" --description "Log
 clean up may have failed which caused the disk to be full" --priority 2 --
source ec2 --operational-data '{"/aws/automations":{"Value":"[{\"automationId
\": \"AWS-ASGEnterStandby\", \"automationType\": \"AWS::SSM::Automation
\"}]","Type":"SearchableString"}}' --notifications Arn="arn:aws:sns:us-
west-2:12345678:TestUser"
```

Windows

```
aws ssm create-ops-item --title "RDS instance not responding" --description
 "RDS instance not responding to ping" --priority 1 --source RDS --
operational-data={\"RDS\":{\"Value\":\"abcd\",\"Type\":\"SearchableString
\"}} --notifications Arn="arn:aws:sns:us-west-1:12345678:TestUser1" --tags
 "Key=RDS,Value=ProductionServers"
```

La commande suivante utilise la clé `/aws/resources` dans OperationalData pour créer un OpsItem avec une ressource associée à l'instance EC2.

```
aws ssm create-ops-item --title "EC2 instance disk full" --description "Log clean
 up may have failed which caused the disk to be full" --priority 2 --source ec2 --
operational-data={\"/aws/resources\":{\"Value\":\"[{\\"""arn\\""":\\"""arn:aws:ec2:us-
east-1:123456789012:instance/i-1234567890abcdef0\\"""}]\",\"Type\":\"SearchableString
\"}}
```

La commande suivante utilise la clé `/aws/automations` dans OperationalData pour créer un OpsItem qui spécifie le document AWS-RestartEC2Instance en tant que runbook Automation associé.

```
aws ssm create-ops-item --title "EC2 instance disk full" --description "Log clean
 up may have failed which caused the disk to be full" --priority 2 --source ec2 --
operational-data={\"/aws/automations\":{\"Value\":\"[{\\"""automationId\\""":\\"""AWS-
RestartEC2Instance\\”""",\\"""automationType\\""":\\"""AWS::SSM::Automation\\"""}]\",
\"Type\":\"SearchableString\"}}
```

Spécifier les données opérationnelles à partir d'un fichier

Lorsque vous créez un OpsItem, vous pouvez spécifier des données opérationnelles à partir d'un fichier. Le fichier doit être un fichier JSON et le contenu du fichier doit utiliser le format suivant :

```
{
  "key_name": {
    "Type": "SearchableString",
    "Value": "Up to 20 KB of data"
  }
}
```

Voici un exemple de .

```
aws ssm create-ops-item --title "EC2 instance disk full" --description "Log clean
 up may have failed which caused the disk to be full" --priority 2 --source ec2 --
operational-data file:///Users/TestUser1/Desktop/OpsItems/opsData.json --notifications
 Arn="arn:aws:sns:us-west-1:12345678:TestUser1" --tags "Key=EC2,Value=Production"
```

> **Note**
>
> Pour de plus amples informations sur la saisie de paramètres au format JSON sur la ligne de commande sur différents systèmes d'exploitation locaux, veuillez consulter Utilisation de guillemets avec des chaînes dans le AWS Command Line Interface Guide de l'utilisateur.

Le système renvoie des informations similaires à celles qui suivent :

```
{
    "OpsItemId": "oi-1a2b3c4d5e6f"
}
```

2. Maintenant, exécutez la commande suivante pour afficher les détails relatifs à la cible de votre OpsItem :

```
aws ssm get-ops-item --ops-item-id ID
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
    "OpsItem": {
        "CreatedBy": "arn:aws:iam::12345678:user/TestUser",
        "CreatedTime": 1558386334.995,
        "Description": "Log clean up may have failed which caused the disk to be full",
        "LastModifiedBy": "arn:aws:iam::12345678:user/TestUser",
        "LastModifiedTime": 1558386334.995,
        "Notifications": [
            {
                "Arn": "arn:aws:sns:us-west-1:12345678:TestUser"
            }
        ],
        "Priority": 2,
        "RelatedOpsItems": [],
        "Status": "Open",
        "OpsItemId": "oi-1a2b3c4d5e6f",
        "Title": "EC2 instance disk full",
        "Source": "ec2",
        "OperationalData": {
            "EC2": {
                "Value": "12345",
                "Type": "SearchableString"
            }
        }
    }
}
```

3. Exécutez la commande suivante pour mettre à jour une tâche OpsItem. Cette commande modifie le statut de `Open` (la valeur par défaut) et le remplace par `InProgress`.

```
aws ssm update-ops-item --ops-item-id ID --status InProgress
```

La commande n'a aucune sortie.

4. Exécutez à nouveau la commande suivante pour vérifier que le statut est passé à `InProgress`

```
aws ssm get-ops-item --ops-item-id ID
```

Utilisation des OpsItems

Cette section décrit comment configurer les options disponibles dans une OpsItem. Pour plus d'informations sur la création d'OpsItems, consultez Création de OpsItems (p. 188).

Rubriques

- Utilisation des ressources connexes (p. 193)
- Modification des détails OpsItem (p. 194)
- Utilisation d'OpsItems connexes et similaires (p. 195)
- Utilisation de données opérationnelles (p. 196)
- Réduction de OpsItems en double (p. 198)

Utilisation des ressources connexes

Une ressource connexe est la ressource impactée (la ressource qui doit faire l'objet d'investigations ou la ressource qui a déclenché l'événement Amazon EventBridge qui a créé l'OpsItem). Chaque OpsItem comporte une section Related resources (Ressources connexes). Si EventBridge crée l'OpsItem, le système remplit automatiquement l'OpsItem avec l'ARN (Amazon Resource Name) de la ressource. Vous pouvez également spécifier manuellement les ARN de ressources connexes. Pour certains types d'ARN, OpsCenter crée automatiquement un lien ciblé qui permet d'afficher des détails sur la ressource sans avoir à consulter d'autres pages de la console pour afficher ces informations. Par exemple, vous pouvez spécifier l'ARN d'une instance EC2. Dans OpsCenter, vous pouvez ensuite consulter tous les détails que Amazon EC2 fournit sur cette instance. Pour afficher une liste des types de ressources qui créent automatiquement des liens ciblés vers les ressources connexes, consultez Référence des ressources prises en charge (p. 206).

> **Note**
>
> Vous pouvez ajouter manuellement les ARN des ressources connexes supplémentaires. Chaque OpsItem peut répertorier un maximum de 100 ARN de ressources connexes.

Pour afficher et ajouter des ressources connexes

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez OpsCenter.
3. Cliquez sur l'onglet OpsItems.
4. Choisissez un ID OpsItem.

5. Pour afficher des informations sur les ressources impactées, choisissez l'onglet Détails des ressources connexes.

Cet onglet affiche des informations sur la ressource de plusieurs services AWS. Développez la section Resource details (Détails sur la ressource) pour afficher des informations sur cette ressource, telles que fournies par le service AWS qui l'héberge. Vous pouvez également basculer entre d'autres ressources connexes associées à cet OpsItem en utilisant la liste Related resources (Ressources connexes).

6. Pour ajouter des ressources connexes, cliquez sur l'onglet Présentation.
7. Dans la section Ressources connexes, choisissez Ajouter.
8. Pour ARN de ressource, entrez l'ARN de la ressource, puis choisissez Ajouter.

   Note

   Si vous ne connaissez pas l'ARN de la ressource, vous pouvez le créer manuellement. Pour plus d'informations sur la création d'un ARN, consultez Amazon Resource Names (ARN) et espaces de noms AWS Service dans le Référence générale d'Amazon Web Services.

Modification des détails OpsItem

La section Détails sur l'OpsItem inclut des informations sur l'OpsItem, notamment la description, le titre, la source, l'ID d'OpsItem et le statut, parmi d'autres informations.

Pour les OpsItems qui ont été créés automatiquement, Amazon EventBridge remplit les champs Titre, Source et Description. Vous pouvez modifier les champs Titre et Description, mais vous ne pouvez pas modifier le champ Source.

À propos du statut OpsItem

Lorsque vous modifiez un OpsItem, vous pouvez spécifier un statut. La liste Statuts inclut les options suivantes :

Statut	Details (Détails)
Ouvert	Actif dans le système, mais pas en cours de traitement par un ingénieur.
En cours	Actif dans le système et en cours de traitement par un ingénieur.
Résolu	Non actif dans le système, mais disponible dans Rechercher et lorsque vous utilisez le filtre Résolu sur la page Présentation de OpsItem. Vous pouvez modifier un OpsItem résolu pour modifier le statut sur Ouvert ou En cours.

Vous pouvez consulter les rapports sur les statuts OpsItem sur l'onglet Récapitulatif. Pour plus d'informations, consultez la section Affichage des rapports de synthèse OpsCenter (p. 206).

À propos de la priorité OpsItem

Lorsque vous modifiez une instance OpsItem, vous pouvez choisir une priorité pou OpsItem en choisissant une valeur comprise entre 1 et 5. Nous recommandons que votre organisation détermine ce que chaque niveau de priorité signifie et un accord de niveau de service correspondant pour chacun d'entre eux.

À propos du champ Notifications

Lorsque vous modifiez un OpsItem, vous pouvez spécifier l'ARN d'une rubrique SNS dans le champ Notifications. En spécifiant un ARN, vous vous assurez que toutes les parties prenantes reçoivent une notification lorsque le OpsItem est modifié, y compris les modifications de statut. Vous pouvez le trouver utile pour créer différents ARN de notifications sur différents types de ressources AWS ou les différents environnements. Pour de plus amples informations, veuillez consulter Amazon Simple Notification Service Manuel du développeur.

> **Important**
>
> La rubrique SNS doit exister dans la même région AWS que le composant OpsItem. S'ils se trouvent dans des régions différentes, le système renvoie une erreur.

Pour modifier les détails de OpsItem

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez OpsCenter.
3. Choisissez un ID OpsItem pour ouvrir la page des détails.
4. Dans la section Détails sur OpsItem, choisissez Modifier.
5. Modifiez les détails de la OpsItem conformément aux exigences et recommandations spécifiées par votre organisation.
6. Lorsque vous avez terminé, choisissez Save.

Utilisation d'OpsItems connexes et similaires

Les fonctions OpsItem connexes et similaires sont conçues pour vous aider à étudier les problèmes d'exploitation tout en fournissant un contexte concernant la portée d'un problème. Dans la section

OpsItems connexes, vous pouvez spécifier un maximum de 10 ID pour d'autres OpsItems pertinents pour le OpsItem en cours. OpsItems peut être lié de différentes façons, dont une relation parent-enfant entre OpsItems, une cause racine ou un doublon.

La fonctionnalité OpsItems similaires est une liste générée par le système des OpsItems qui peuvent être liés ou vous intéresser. Pour générer la liste, le système analyse les titres et les descriptions de tous les OpsItems et renvoie les OpsItems qui utilisent des mots similaires.

Pour ajouter un OpsItem à partir d'OpsItems similaires

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez OpsCenter.
3. Choisissez un ID OpsItem pour ouvrir la page des détails.
4. Dans la section OpsItem connexes, choisissez Ajouter.
5. Pour OpsItem ID, spécifiez un identificateur.
6. Choisissez Add (Ajouter).

Utilisation de données opérationnelles

Les données opérationnelles sont des données personnalisées qui fournissent des détails sur le OpsItem. Par exemple, vous pouvez spécifier les fichiers journaux, les chaînes d'erreur, les clés de licence, les conseils de dépannage ou d'autres données pertinentes. Vous saisissez les données opérationnelles sous forme de paires clé-valeur. La clé possède une longueur maximale de 128 caractères. La valeur a une taille maximale de 20 Ko. Vous pouvez saisir plusieurs paires clé/valeur pour les données opérationnelles.

> **Important**
>
> Les clés des données opérationnelles ne peuvent pas commencer par les éléments suivants : amazon, aws, amzn, ssm, /amazon, /aws, /amzn, /ssm.

Vous pouvez choisir de rendre les données interrogeables par d'autres utilisateurs dans le compte ou vous pouvez limiter l'accès à la recherche. Les données interrogeables signifient que tous les utilisateurs ayant accès à la page Présentation d'OpsItem (telle que fournie par l'action d'API Décrire les OpsItem) peuvent afficher et faire des recherches sur les données spécifiées. Les données opérationnelles qui ne sont pas interrogeables sont uniquement visibles par les utilisateurs qui ont accès à l'OpsItem (comme fourni par l'action d'API GetOpsItem).

Pour ajouter des données opérationnelles à un OpsItem

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez OpsCenter.
3. Choisissez un ID OpsItem pour ouvrir la page des détails.
4. Développez l'une ou l'autre des Données opérationnelles.
5. S'il n'existe aucune donnée opérationnelle pour OpsItem, choisissez Ajouter. Si des données opérationnelles existent déjà pour le OpsItem, choisissez Gérer.
6. Pour Clé, spécifiez un ou plusieurs mots pour aider les utilisateurs à comprendre l'objectif des données. La clé ne peut pas commencer par les éléments suivants : amazon, aws, amzn, ssm, /amazon, /aws, /amzn, /ssm.
7. Dans Valeur, précisez les données.
8. Choisissez Save.

Une fois que vous avez créé les données opérationnelles, vous pouvez modifier la clé et la valeur, supprimer les données opérationnelles ou ajouter d'autres paires clé-valeur en choisissant Gérer.

Note

Vous pouvez filtrer les OpsItems en utilisant l'opérateur Données opérationnelles sur la page OpsItems. Dans la zone Search (Rechercher), choisissez Operational data (Données opérationnelles), puis saisissez une paire clé-valeur au format JSON. Vous devez saisir la paire clé-valeur en utilisant le format suivant : `{"key":"key_name","value":"a_value"}`

Réduction de OpsItems en double

OpsCenter utilise une combinaison de logique intégrée et de chaînes de déduplication configurables pour éviter la création de OpsItems en double. La logique de déduplication intégrée est appliquée chaque fois que l'action d'API CréerOpsItem est appelée. Lorsque vous créez l'OpsItem, Systems Manager crée et stocke un hachage en fonction de la chaîne de déduplication et de la ressource qui a déclenché l'OpsItem. Lorsqu'une demande est faite pour créer un nouvel OpsItem, le système vérifie la chaîne de déduplication de la nouvelle demande. Si un hachage correspondant existe pour cette chaîne de déduplication, Systems Manager ne crée pas de nouvel OpsItem.

Notez les informations suivantes sur OpsCenter et la déduplication :

- Les chaînes de déduplication ne sont pas sensibles à la casse. Si le système trouve une correspondance dans une chaîne de déduplication OpsItem, quelle que soit la casse, le nouveau OpsItem n'est pas créé.
- Si le système trouve une correspondance dans une chaîne de déduplication OpsItem, et que OpsItem a l'état `Open`, le nouveau OpsItem n'est pas créé. Si une correspondance est trouvée dans une chaîne de déduplication OpsItem qui a l'état `Resolved`, le système crée une nouveau OpsItem.
- Si le système trouve une correspondance dans une chaîne de déduplication OpsItem, mais que les ressources sont différentes, le système crée le nouveau OpsItem.

Configuration des chaînes de déduplication

OpsCenter inclut les options suivantes pour configurer les chaînes de déduplication.

- Modifier les chaînes de déduplication préconfigurées: Chacun des OpsItem par défaut EventBridge Les règles incluent une chaîne de déduplication préconfigurée. Vous pouvez modifier ces chaînes de déduplication dans EventBridge.
- Spécifier manuellement les chaînes de déduplication: Vous pouvez entrer une chaîne de déduplication à l'aide de l'option Chaîne de déduplication dans la console ou l' `OperationalData` lorsque vous créez un nouveau OpsItem en utilisant soit le AWS CLI ou Outils AWS pour Windows PowerShell.

Une fois que le système a créé un OpsItem, il remplit le champ Chaîne de déduplication si une chaîne a été spécifiée. Voici un exemple :

Une fois que vous avez créé un OpsItem, vous ne pouvez pas modifier ni changer les chaînes de déduplication dans cet OpsItem.

Cette section comprend les procédures suivantes pour configurer les chaînes de déduplication.

- Modification d'une chaîne de déduplication dans une règle EventBridge par défaut de OpsCenter (p. 199)
- Spécification d'une chaîne de déduplication à l'aide de l'AWS CLI (p. 200)

> **Note**
>
> Pour plus d'informations sur la saisie des chaînes de déduplication lorsque vous créez manuellement un OpsItem dans la console, consultez Création manuelle de OpsItems (p. 189).

Modification d'une chaîne de déduplication dans une règle EventBridge par défaut de OpsCenter

Utilisez la procédure suivante pour spécifier une chaîne de déduplication pour un EventBridge règle qui cible OpsCenter.

Pour modifier une chaîne de déduplication dans une règle EventBridge par défaut OpsItem

1. Connectez-vous à AWS Management Console et ouvrez la console Amazon EventBridge à l'adresse https://console.aws.amazon.com/events/.
2. Dans le volet de navigation, choisissez Rules.
3. Choisissez une règle, puis choisissez Modifier.
4. Dans le Sélectionner les cibles section, développer Configurer l'entrée. Dans la partie inférieure Transformateur d'entrée , localisez le champ `"operationalData": { "/aws/dedup"` Entrée JSON et chaînes de déduplication que vous souhaitez modifier.

   La chaîne de déduplication dans les règles d'entrée EventBridge utilise le format JSON suivant.

   ```
   "operationalData": { "/aws/dedup": {"type": "SearchableString","value": "{\"dedupString
   \":\"Words the system should use to check for duplicate OpsItems\"}"}
   ```

   Voici un exemple de .

   ```
   "operationalData": { "/aws/dedup": {"type": "SearchableString","value": "{\"dedupString
   \":\"SSMOpsCenter-EBS-volume-performance-issue\"}"}
   ```

5. Modifiez les chaînes de déduplication, puis choisissez Mettre à jour pour terminer la mise à jour de la règle.

Spécification d'une chaîne de déduplication à l'aide de l'AWS CLI

Vous pouvez spécifier une chaîne de déduplication lorsque vous créez manuellement un nouvel OpsItem l'aide de l'AWS CLI. Vous indiquez la chaîne de déduplication à l'aide du parmètre `OperationalData` Le paramètre utilise la syntaxe JSON, comme illustré ici.

```
--operational-data '{"/aws/dedup":{"Value":"{\"dedupString\": \"Words the system should use
 to check for duplicate OpsItems\"}","Type":"SearchableString"}}'
```

Voici un exemple de commande qui spécifie une chaîne de déduplication de `disk full`.

Linux

```
aws ssm create-ops-item \
    --title "EC2 instance disk full" \
    --description "Log clean up may have failed which caused the disk to be full" \
    --priority 1 \
    --source ec2 \
    --operational-data '{"/aws/dedup":{"Value":"{\"dedupString\": \"disk full
\"}","Type":"SearchableString"}}' \
    --tags "Key=EC2,Value=ProductionServers" \
    --notifications Arn="arn:aws:sns:us-west-1:12345678:TestUser"
```

Windows

```
aws ssm create-ops-item ^
    --title "EC2 instance disk full" ^
    --description "Log clean up may have failed which caused the disk to be full" ^
    --priority 1 ^
    --source EC2 ^
    --operational-data={\"/aws/dedup\":{\"Value\":\"{\\"""dedupString\\""":\\"""disk
 full\\"""}\",\"Type\":\"SearchableString\"}} ^
    --tags "Key=EC2,Value=ProductionServers" --notifications Arn="arn:aws:sns:us-
west-1:12345678:TestUser"
```

Correction des problèmes OpsItem avec Systems Manager Automation

L'automatisation AWS Systems Manager vous aide à corriger rapidement les problèmes avec les ressources AWS identifiées dans votre OpsItems. L'automatisation utilise les documents d'automatisation SSM prédéfinis (runbooks) pour corriger des problèmes communs avec des ressources AWS. Par exemple, l'automatisation inclut des runbooks pour effectuer les actions suivantes :

- Arrêter, démarrer, redémarrer et résilier Amazon Relational Database Service (Amazon RDS) et Amazon Elastic Compute Cloud (instances EC2).
- Créer des ressources AWS telles que les Amazon Machine Images (AMI), les instantanés Amazon Elastic Block Store (Amazon EBS) et les sauvegardes Amazon DynamoDB.
- Configurez une ressource pour utiliser les services AWS, y compris la journalisation et le contrôle de version des compartiments Amazon EventBridge, AWS CloudTrail, et Amazon Simple Storage Service (Amazon S3).
- Attachez un profil d'instance AWS Identity and Access Management (IAM) à une instance.
- Résolution des problèmes de connectivité SSH et RDP pour les instances EC2.

- Réinitialisez l'accès à une instance EC2.

Chaque OpsItem dans le AWS Management Console comprend un Livres d'exécution comme illustré dans l'exemple suivant.

Fonctions du document d'automatisation dans OpsCenter

La liste suivante décrit certaines des fonctionnalités disponibles pour vous aider à exécuter des documents Automation et à corriger les problèmes.

- Lorsque vous choisissez une ressource AWS qui a généré un OpsItem, OpsCenter affiche une liste des documents Automation que vous pouvez exécuter sur cette ressource, comme illustré dans l'exemple suivant.

- Lorsque vous choisissez un document Automation dans la liste, OpsCenter préremplit certains des champs requis pour exécuter le document. Dans l'exemple suivant, OpsCenter a prérempli le ID du volume champ.

- OpsCenter conserve un enregistrement de 30 jours des documents Automation exécutés pour un OpsItem, comme illustré dans l'exemple suivant.

- Dans le Statut et résultats , vous pouvez choisir un statut pour afficher des détails importants sur cette exécution, tels que la raison pour laquelle une automatisation a échoué et l'étape du document d'automatisation qui était en cours d'exécution lorsque l'échec s'est produit, comme illustré dans l'exemple suivant.

- Le Détails des ressources connexes pour un sélectionné OpsItem inclut le Exécuter l'automatisation liste. La liste vous permet de choisir des documents Automation récents ou spécifiques aux ressources que vous pouvez exécuter pour corriger les problèmes. Cette page comprend également des fournisseurs de données utiles, notamment Amazon CloudWatch les métriques et les alarmes, AWS CloudTrail et les détails de AWS Config, pour n'en citer que quelques-uns.

- Vous pouvez afficher des informations sur un document Automation en choisissant son nom dans la console ou en utilisant l' Référence sur les détails du document Automation (p. 511).

Utilisation d'un runbook pour corriger un problème OpsItem

Lorsque vous exécutez un Systems Manager Document d'automatisation (runbook) à partir d'un OpsItem, vous pouvez exécuter une version simple ou vous pouvez choisir le Configuration avancée option. La Configuration avancée ouvre le runbook dans Automatisation Systems Manager, qui fournit plusieurs options pour l'exécution du runbook.

La procédure suivante décrit comment exécuter une version simple d'un runbook. Pour de plus amples sur l'exécution d'un runbook Configuration avancée, veuillez consulter Utilisation des automatisations (p. 313).

Avant de commencer

Avant d'exécuter un document d'automatisation (runbook) pour corriger un OpsItem , procédez comme suit :

- Vérifiez que vous avez l'autorisation d'exécuter des documents d'automatisation Systems Manager. Pour plus d'informations, consultez la section Configuration d'Automation (p. 308).
- Collectez les informations d'ID spécifiques à la ressource pour l'automatisation que vous souhaitez exécuter. Par exemple, si vous souhaitez exécuter une automatisation qui redémarre une instance EC2, vous devez spécifier l'ID de l'instance pour redémarrer.

Pour exécuter un document Automation (runbook) pour corriger un OpsItem problème

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez OpsCenter.
3. Choisissez l'ID OpsItem pour ouvrir la page des détails.

4. Faites défiler jusqu'à la section Runbooks.
5. Utilisez la barre de recherche Runbooks ou les numéros en haut à droite pour trouver le document Automation que vous souhaitez exécuter.

6. Choisissez un runbook, puis choisissez Exécuter.
7. Entrez les informations requises pour le runbook, puis choisissez Exécuter.
8. Dans le panneau de navigation, choisissez Automatisation, puis sélectionnez le lien ID d'exécution pour afficher les étapes et le statut de l'exécution.

Utilisation de runbooks associés

Après avoir exécuté un document Automation (runbook) à partir d'un OpsItem, le runbook est automatiquement associé à la ressource associée de ce OpsItem pour référence ultérieure. Les runbooks associés ont un niveau supérieur aux autres dans la liste Runbooks, comme indiqué ci-après.

Les runbooks associés sont également disponibles dans la liste Exécuter l'automatisation dans la section Ressources connexes, comme indiqué ci-après.

Utilisez la procédure suivante pour exécuter un document Automation (runbook) qui a déjà été associé à une ressource associée dans un OpsItem. Pour plus d'informations sur l'ajout de ressources connexes, consultez Utilisation des OpsItems (p. 193).

Pour exécuter un runbook associé à une ressource afin de corriger un OpsItem problème

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.

2. Dans le panneau de navigation, choisissez OpsCenter.
3. Ouvrez OpsItem.
4. Dans le Ressources connexes , choisissez la ressource sur laquelle vous souhaitez exécuter le document Automation (runbook).
5. Choisir Exécuter l'automatisation, puis choisissez le document Automation associé que vous souhaitez exécuter.
6. Entrez les informations requises pour le runbook, puis choisissez Exécuter.
7. Dans le panneau de navigation, choisissez Automatisation, puis sélectionnez le lien ID d'exécution pour afficher les étapes et le statut de l'exécution.

Affichage des rapports de synthèse OpsCenter

OpsCenter inclut une page récapitulative qui affiche automatiquement les informations suivantes :

- Résumé de statut de l'OpsItem : un récapitulatif de l'OpsItems par statut (Ouvert, En cours, Résolu, Ouvert et En cours).
- Sources avec la plupart des OpsItems ouverts : une répartition des principaux services AWS qui comportent un OpsItems ouvert.
- OpsItems par source et âge : nombre d'OpsItems regroupés par source et jours depuis la création.

Pour afficher la page de synthèse de OpsCenter

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez OpsCenter.
3. Sur la page Présentation de OpsItems, choisissez Récapitulatif.
4. Sous OpsItems par source et âge, choisissez la barre de recherche pour filtrer les OpsItems en fonction de la Source. Utilisez la liste pour filtrer en fonction de l'état.

Référence des ressources prises en charge

OpsCenter crée automatiquement un lien ciblé pour la page de ressource principale lorsque vous spécifiez l'Amazon Resource Name (ARN) pour les types suivants de ressources AWS. Par exemple, si vous spécifiez l'ARN d'une instance EC2 dans le champ Ressources connexes, OpsCenter crée un lien ciblé pour les informations sur cette instance dans la console Amazon EC2. Cela vous permet d'afficher des informations détaillées sur vos ressources AWS qui peuvent être affectés sans avoir à quitter OpsCenter. Pour plus d'informations sur l'ajout de ressources connexes, consultez Utilisation des ressources connexes (p. 193).

Ressources prises en charge

Nom de la ressource	Format ARN
Certificat AWS Certificate Manager	`arn:aws:acm:region:account-id:certificate/certificate-id`
Groupe Amazon EC2 Auto Scaling	`arn:aws:autoscaling:region:account-id:autoScalingGroup:groupid:autoScalingGroupName/groupfriendlyname`

Nom de la ressource	Format ARN
Amazon CloudFront distribution	`arn:aws:cloudfront::account-id:*`
Pile AWS CloudFormation	`arn:aws:cloudformation:region:account-id:stack/stackname/additionalidentifier`
Alarme Amazon CloudWatch	`arn:aws:cloudwatch:region:account-id:alarm:alarm-name`
Journal de suivi AWS CloudTrail	`arn:aws:cloudtrail:region:account-id:trail/trailname`
Projet AWS CodeBuild	`arn:aws:codebuild:region:account-id:resourcetype/resource`
AWS CodePipeline	`arn:aws:codepipeline:region:account-id:resource-specifier`
Table Amazon DynamoDB	`arn:aws:dynamodb:region:account-id:table/tablename`
Passerelle client Amazon EC2	`arn:aws:ec2:region:account-id:customer-gateway/cgw-id`
Adresse IP elastic Amazon EC2	`arn:aws:ec2:region:account-id:eip/eipalloc-id`
Hôte dédié Amazon EC2	`arn:aws:ec2:region:account-id:dedicated-host/host-id`
Instance Amazon EC2	`arn:aws:ec2:region:account-id:instance/instance-id`
Passerelle Internet Amazon EC2	`arn:aws:ec2:region:account-id:internet-gateway/igw-id`
Liste de contrôle d'accès (ACL) réseau Amazon EC2	`arn:aws:ec2:region:account-id:network-acl/nacl-id`
Interface réseau Amazon EC2	`arn:aws:ec2:region:account-id:network-interface/eni-id`
Table de routage Amazon EC2	`arn:aws:ec2:region:account-id:route-table/route-table-id`

Nom de la ressource	Format ARN
Groupe de sécurité Amazon EC2	`arn:aws:ec2:region:account-id:security-group/security-group-id`
Sous-réseau Amazon EC2	`arn:aws:ec2:region:account-id:subnet/subnet-id`
Volume Amazon EC2	`arn:aws:ec2:region:account-id:volume/volume-id`
VPC Amazon EC2	`arn:aws:ec2:region:account-id:vpc/vpc-id`
Connexion VPN Amazon EC2	`arn:aws:ec2:region:account-id:vpn-connection/vpn-id`
Passerelle VPN Amazon EC2	`arn:aws:ec2:region:account-id:vpn-gateway/vgw-id`
Application AWS Elastic Beanstalk	`arn:aws:elasticbeanstalk:region:account-id:application/applicationname`
Elastic Load Balancing (équilibreur de charge classique)	`arn:aws:elasticloadbalancing:region:account-id:loadbalancer/name`
Elastic Load Balancing (équilibreur de charge d'application)	`arn:aws:elasticloadbalancing:region:account-id:loadbalancer/app/load-balancer-name/load-balancer-id`
Elastic Load Balancing (équilibreur de charge du réseau)	`arn:aws:elasticloadbalancing:region:account-id:loadbalancer/net/load-balancer-name/load-balancer-id`
AWS Identity and Access Management (IAM) groupe	`arn:aws:iam::account-id:group/group-name`
Stratégie IAM	`arn:aws:iam::account-id:policy/policy-name`
IAM rôle	`arn:aws:iam::account-id:role/role-name`
Utilisateur IAM	`arn:aws:iam::account-id:user/user-name`
Fonction AWS Lambda	`arn:aws:lambda:region:account-id:function:function-name`

Nom de la ressource	Format ARN
Cluster Amazon Relational Database Service (Amazon RDS)	`arn:aws:rds:region:account-id:cluster:db-cluster-name`
Instance de base de données Amazon RDS	`arn:aws:rds:region:account-id:db:db-instance-name`
Abonnement Amazon RDS	`arn:aws:rds:region:account-id:es:subscription-name`
Groupe de sécurité Amazon RDS	`arn:aws:rds:region:account-id:secgrp:security-group-name`
Instantané de cluster Amazon RDS	`arn:aws:rds:region:account-id:cluster-snapshot:cluster-snapshot-name`
Groupe de sous-réseaux Amazon RDS	`arn:aws:rds:region:account-id:subgrp:subnet-group-name`
cluster Amazon Redshift	`arn:aws:redshift:region:account-id:cluster:cluster-name`
Groupe de paramètres Amazon Redshift	`arn:aws:redshift:region:account-id:parametergroup:parameter-group-name`
Groupe de sécurité Amazon Redshift	`arn:aws:redshift:region:account-id:securitygroup:security-group-name`
Instantané de cluster Amazon Redshift	`arn:aws:redshift:region:account-id:snapshot:cluster-name/snapshot-name`
Groupe de sous-réseaux Amazon Redshift	`arn:aws:redshift:region:account-id:subnetgroup:subnet-group-name`
Compartiment Amazon Simple Storage Service (Amazon S3)	`arn:aws:s3:::bucket_name`
Enregistrement AWS Config de l'inventaire AWS Systems Manager des instances gérées	`arn:aws:ssm:region:account-id:managed-instance-inventory/instance_id`
Association Systems Manager State Manager	`arn:aws:ssm:region:account-id:association/association_ID`

Audit et journalisation de l'activité de OpsCenter

AWS CloudTrail capture les appels d'API OpsCenter effectués dans la console AWS Systems Manager, l'AWS CLI et le kit SDK Systems Manager. Vous pouvez voir les informations contenues dans la console CloudTrail ou dans un compartiment Amazon Simple Storage Service (Amazon S3). Un seul compartiment est utilisé pour tous les journaux CloudTrail de votre compte.

Les journaux d'actionsOpsCenter montrent les activités OpsItem de création, de mise à jour, d'obtention et de description. Pour de plus amples informations sur l'affichage et l'utilisation des journaux CloudTrail de l'activité Systems Manager, veuillez consulter Journalisation des appels d'API AWS Systems Manager avec AWS CloudTrail (p. 1336).

Tableaux de bord Amazon CloudWatch hébergés par Systems Manager

Les tableaux de bord Amazon CloudWatch sont des pages d'accueil personnalisables de la console CloudWatch que vous pouvez utiliser pour surveiller vos ressources dans une seule vue, même les ressources réparties dans différentes régions. Vous pouvez utiliser les tableaux de bord CloudWatch pour créer des vues personnalisées des métriques et alarmes de vos ressources AWS. Avec les tableaux de bord, vous pouvez créer les éléments suivants :

- Une vue unique pour les métriques et alarmes sélectionnées afin de vous aider à évaluer l'état de vos ressources et applications d'une ou de plusieurs régions. Vous pouvez sélectionner la couleur utilisée pour chaque métrique sur chaque graphique afin de suivre facilement la même métrique sur plusieurs graphiques.
- Un manuel opérationnel qui fournit des conseils aux membres d'une équipe pendant des événements opérationnels sur la façon de répondre à des incidents spécifiques.
- Une vue globale des mesures critiques des ressources et applications qui peut être partagée par les membres de l'équipe pour un flux de communication plus rapide au cours des événements opérationnels.

Vous pouvez créer des tableaux de bord à l'aide de la console, de l'AWS CLI ou de l'API PutDashboard. Pour plus d'informations, consultez Utilisation des tableaux de bord Amazon CloudWatch dans le Guide de l'utilisateur Amazon CloudWatch.

Tableaux de bord Personal Health et Trusted Advisor hébergés par Systems Manager

Systems Manager héberge deux outils en ligne pour vous aider à mettre en service vos ressources et surveiller votre compte pour les événements de santé. Trusted Advisor est un outil en ligne qui vous fournit des conseils en temps réel pour vous aider à mettre en service vos ressources conformément aux bonnes pratiques AWS. Pour plus d'informations, consultez la section Trusted Advisor.

Le AWS Personal Health Dashboard fournit des informations sur les événements AWS Health susceptibles d'affecter votre compte. Les informations sont présentées de deux manières : un tableau de bord qui montre les événements récents et à venir organisés par catégorie, et un journal des événements complet qui contient tous les événements des 90 derniers jours. Pour plus d'informations, consultez Démarrer avec le AWS Personal Health Dashboard.

Important

Trusted Advisor et AWS Personal Health Dashboard affichent des données dans Systems Manager uniquement dans les régions AWS suivantes.

Nom de la région	Région
USA Est (Ohio)	us-east-2
USA Est (Virginie du Nord)	us-east-1
USA Ouest (Californie du Nord)	us-west-1
USA Ouest (Oregon)	us-west-2
Asie-Pacifique (Mumbai)	ap-south-1
Asie-Pacifique (Osaka-Local)	ap-northeast-3
Asie-Pacifique (Séoul)	ap-northeast-2
Asie-Pacifique (Singapour)	ap-southeast-1
Asie-Pacifique (Sydney)	ap-southeast-2
Asie-Pacifique (Tokyo)	ap-northeast-1
Canada (Centre)	ca-central-1
Europe (Francfort)	eu-central-1
Europe (Irlande)	eu-west-1
Europe (Londres)	eu-west-2
Europe (Paris)	eu-west-3
Europe (Stockholm)	eu-north-1
Amérique du Sud (São Paulo)	sa-east-1

AWS Systems Manager Application Management

Application Management est une suite de fonctionnalités qui vous aident à gérer vos applications exécutées dans AWS.

Rubriques

- Resource Groups dans AWS Systems Manager (p. 212)
- AWS AppConfig (p. 218)
- AWS Systems Manager Parameter Store (p. 218)

Resource Groups dans AWS Systems Manager

Vous pouvez utiliser des groupes de ressources pour organiser vos ressources AWS. Les groupes de ressources facilitent la gestion, le contrôle et l'automatisation des tâches simultanément sur un grand nombre de ressources.

Groupes de ressources AWS fournit deux méthodes générales pour définir un groupe de ressources. Les deux méthodes impliquent l'utilisation d'une requête pour identifier les membres d'un groupe.

La première méthode est basée sur des balises appliquées aux ressources AWS pour ajouter des ressources à un groupe. En utilisant cette méthode, vous appliquez les mêmes paires de balises clé/valeur aux ressources de différents types de votre compte, puis vous utilisez le service Groupes de ressources AWS pour créer un groupe basé sur cette paire de balises.

La deuxième méthode est basée sur les ressources disponibles dans une pile AWS CloudFormation individuelle. Avec cette méthode, vous choisissez une pile AWS CloudFormation, puis les types de ressources dans la pile dont vous souhaitez qu'elles fassent partie du groupe.

Pour plus d'informations sur ces méthodes, consultez Générez des requêtes et des groupes dans Groupes de ressources AWS, dans le Guide de l'utilisateur Groupes de ressources AWS.

Pour être ajoutées à un groupe de ressources, les ressources doivent toutes se trouver dans la même région AWS. Les ressources doivent également être l'un des types de ressources pris en charge pour une utilisation dans les groupes de ressources. Par exemple, dans le service AWS Identity and Access Management (IAM), vous pouvez appliquer des balises aux utilisateurs et aux rôles, mais seuls ces derniers sont pris en charge par les groupes de ressources. Dans le Amazon Simple Notification Service (Amazon SNS), vous pouvez ajouter des balises aux rubriques pour les ajouter à un groupe de ressources, mais vous ne pouvez pas ajouter de balises aux abonnements. Dans AWS Systems Manager, vous pouvez appliquer des balises à plusieurs types de ressources, y compris des fenêtres de maintenance, des instances gérées et des références de correctifs, mais pas pour modifier des calendriers ou des packages Distributor.

Vous pouvez ajouter des balises à vos ressources AWS de différentes manières :

- Lorsque vous créez la ressource
- Lorsque vous mettez à jour la ressource

- Utiliser Tag Editor (Éditeur de balises) dans le service Resource Groups

Types de ressource AWS pris en charge

Pour obtenir la liste de tous les services avec des types de ressources qui peuvent être ajoutés à des groupes de ressources à l'aide de balises et des ressources qu'ils prennent en charge, consultez Ressources prises en charge dans le Guide de l'utilisateur Groupes de ressources AWS.

Types de ressource Systems Manager pris en charge :

Les types de ressources Systems Manager que vous pouvez baliser pour les ajouter à des groupes de ressources sont les suivants :

- Documents SSM
- Instances gérées
- Fenêtres de maintenance
- Paramètres SSM
- Références de correctifs

Autorisations pour travailler avec des groupes de ressources et l'éditeur de balises

Avant que les utilisateurs de votre compte AWS puissent utiliser des groupes de ressources et des balises dans le service Resource Groups et l'Éditeur de balises, un utilisateur disposant d'un accès administrateur doit leur fournir les autorisations nécessaires. Pour de plus amples informations sur l'octroi aux utilisateurs Systems Manager de votre compte l'accès à Resource Groups et à Tag Editor dans AWS Management Console et la fonctionnalité de groupes de ressources dans Systems Manager, veuillez consulter Créer des groupes d'utilisateurs (p. 31).

Utilisation de l'éditeur de balises

L'utilisation de l'éditeur de balises est le moyen le plus efficace d'ajouter de nombreux types de ressources à un groupe de ressources. Vous pouvez afficher tous les types de ressources pris en charge dans votre compte à partir de la même page. Pour les ressources de certains types, choisissez uniquement celles que vous souhaitez ajouter au groupe de ressources et ajoutez-leur les balises en bloc. Pour plus d'informations sur l'utilisation de l'éditeur de balises, consultez Rechercher des ressources à baliser et Gérer les balises dans le Guide de l'utilisateur Groupes de ressources AWS.

De quelle autre façon puis-je utiliser des ressources balisées ?

L'ajout de ressources à un groupe de ressources n'est que l'une des principales utilisations des balises de ressources. Vous pouvez également utiliser des balises pour spécifier des ressources avec certaines balises appliquées comme cibles d'opérations AWS. Vous pouvez rechercher des ressources avec les mêmes balises qui leur sont appliquées. Vous pouvez créer des stratégies IAMpour accorder ou refuser aux utilisateurs l'accès aux ressources qui sont balisées avec ces balises.

Que puis-je faire avec les groupes de ressources ?

Plusieurs services AWS, y compris Systems Manager, vous permettent d'agir sur, de contrôler ou de partager des ressources AWS en tant que groupe. Par exemple, dans Amazon CloudWatch, vous pouvez concentrer votre affichage pour indiquer les mesures et alarmes d'un seul groupe de ressources. Dans AWS Resource Access Manager, vous pouvez partager les ressources AWS que vous avez ajoutées à un groupe de ressources, en tant que groupe, avec d'autres comptes de votre choix.

Pour plus d'informations sur tous les services AWS qui peuvent utiliser des groupes de ressources, consultez Intégration des services avec Groupes de ressources AWS dans le Guide de l'utilisateur Groupes de ressources AWS.

Dans Systems Manager, vous pouvez utiliser des groupes de ressources de différentes façons.

Tout d'abord, vous pouvez créer et gérer des groupes de ressources. Systems Manager comprend une console qui offre les mêmes fonctionnalités que la console du service Resource Groups. Pour accéder à cette console, procédez comme suit :

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Resource Groups.

Pour plus d'informations sur l'utilisation de la console Resource Groups, consultez Mise en route avec Groupes de ressources AWS dans le Guide de l'utilisateur Groupes de ressources AWS.

Ensuite, dans la fonction Inventaire, vous pouvez sélectionner les instances gérées dont vous souhaitez afficher les données de conformité, en spécifiant un groupe de ressources auquel elles appartiennent.

Enfin, vous pouvez spécifier un groupe de ressources comme cible dans les cas suivants :

- Une commande que vous exécutez dans Fonctionnalité Exécuter la commande.
- Un flux de travail d'automatisation que vous exécutez dans Automation.
- Une association que vous créez dans State Manager.
- Une fenêtre de maintenance que vous créez dans Fenêtres de maintenance.
- Une installation de package ou une opération de mise à jour dans Distributor.

Rubriques

- Affichage des données d'opérations pour Groupes de ressources AWS (p. 214)

Articles de blog AWS et autres

Consultez les articles de blog AWS suivants pour plus d'informations sur les groupes de ressources.

- Utiliser les stratégies de balises pour gérer les balises sur plusieurs comptes AWS (26 NOV 2019)
- Nouveau gestionnaire d'accès aux ressources AWS — Partage de ressources entre comptes (21 NOV 2018)
- Simplify granting access to your AWS resources by using tags on AWS IAM users and roles (19 nov. 2018)
- The Resource Groups Tagging API Makes It Easier to List Your Resources by Using a New Pagination Parameter (22 mai 2017)
- Centralisez la gestion des balises et recherchez des ressources dans l'ensemble des services AWS grâce à la nouvelle API de balisage des groupes de ressources (30 mars 2017)
- Now Organize Your AWS Resources by Using up to 50 Tags per Resource (15 août 2016, mise à jour le 28 déc. 2017)

Affichage des données d'opérations pour Groupes de ressources AWS

Dans le AWS Systems Manager console, le Groupes de ressources AWS affiche les données d'opérations d'un groupe sélectionné sous cinq onglets : Détails, Config., CloudTrail, Surveillance, et Éléments d'opérations.

> **Note**
>
> Ces onglets ne sont pas disponibles lors de l'affichage d'un groupe dans la console Resource Groups.

Les informations de ces onglets peuvent vous aider à comprendre quelles ressources au sein d'un groupe sont conformes et fonctionnent correctement, et quelles ressources demandent une action. Si vous devez agir sur une ressource, vous pouvez utiliser les runbooks d'automatisation Systems Manager pour effectuer des tâches courantes de maintenance et de dépannage.

Onglet	Details (Détails)
Details (Détails)	Pour les groupes de ressources basés sur des balises, cet onglet inclut une vue d'ensemble du groupe, les balises sur lesquelles le groupe est basé et une liste des ressources découvertes par la requête du groupe de ressources. Pour les groupes AWS CloudFormation basés sur une pile, cet onglet inclut des informations détaillées sur la pile AWS CloudFormation, y compris les événements de pile. Pour les groupes de ressources basés sur des balises et sur des piles AWS CloudFormation, si une ressource de la section Group resources (Ressources de groupe) inclut un lien, vous pouvez cliquer sur ce dernier pour afficher des détails supplémentaires sur la ressource sélectionnée. Toutes les ressources AWS ne sont pas incluses dans l'affichage de détails supplémentaires. Pour plus d'informations, consultez Ressources prises en charge dans l'affichage des détails supplémentaires (p. 216) dans cette rubrique.

Onglet	Details (Détails)
Config	Cet onglet contient des informations sur la conformité AWS Config et l'historique des ressources du groupe sélectionné.
CloudTrail	Cet onglet répertorie les événements AWS CloudTrail pour les ressources du groupe sélectionné.
Surveillance	Cet onglet répertorie les alarmes Amazon CloudWatch des ressources du groupe sélectionné. Cet onglet affiche également les tableaux de bord CloudWatch associés à votre groupe de ressources. Si aucune alarme n'a été configurée et aucun tableau de bord n'a été créé pour le groupe de ressources sélectionné, l'onglet est vide.
OpsItems	Cet onglet répertorie tous les éléments de travail opérationnels (Opsitems) pour les ressources AWS CloudFormation basées sur une pile dans le groupe sélectionné. Les OpsItems pour les ressources basées sur des balises ne sont pas pris en charge. Pour les ressources basées sur des balises, la table OpsItems est vide. La table est également vide si aucun OpsItems n'a été créé pour les ressources AWS CloudFormation de ce groupe.

Important

Avant de pouvoir afficher des Opsitems dans l'onglet Opsitems vous devez configurer AWS Systems Manager Explorer. Pour plus d'informations, consultez Mise en route avec Systems Manager Explorer et OpsCenter dans le Guide de l'utilisateur AWS Systems Manager. Après avoir configuré l'Explorateur, Systems Manager remplit automatiquement l'onglet Opsitems.

Ressources prises en charge dans l'affichage des détails supplémentaires

L'onglet Details (Détails) des groupes basés sur des balises et des groupes AWS CloudFormation basés sur des piles comprend une section Group resources (Ressources de groupe). Cette section répertorie chaque ressource du groupe sélectionné.

Pour les groupes de ressources AWS CloudFormation basés sur des piles et les groupes de ressources basés sur des balises, si l'une des ressources de cette section inclut un lien, vous pouvez cliquer dessus pour afficher des détails supplémentaires sur la ressource sélectionnée. La page de détails supplémentaires inclut des données opérationnelles provenant de plusieurs services AWS pour vous aider à évaluer rapidement l'état d'une ressource et les actions récentes qui lui ont été appliquées.

Toutes les ressources AWS ne sont pas incluses dans l'affichage de détails supplémentaires. La liste suivante répertorie les services AWS pris en charge. Lorsqu'elles sont répertoriées dans la section Group ressources (Ressources de groupe), ces ressources incluent un lien vers la page de détails supplémentaires.

```
ACM::Certificate
AutoScaling::AutoScalingGroup
CloudFront::Distribution
CloudFormation::Stack
CloudWatch::Alarm
CloudTrail::Trail
CodeBuild::Project
CodePipeline::Pipeline
DynamoDB::Table
EC2::CustomerGateway
EC2::EIP
EC2::Host
EC2::Instance
EC2::InternetGateway
EC2::NetworkAcl
EC2::NetworkInterface
EC2::RouteTable
EC2::SecurityGroup
EC2::Subnet
```

```
EC2::Volume
EC2::VPC
EC2::VPNConnection
EC2::VPNGateway
ElasticBeanstalk::Application
ElasticLoadBalancing::LoadBalancer
ElasticLoadBalancingV2::NetworkLoadBalancer
ElasticLoadBalancingV2::ApplicationLoadBalancer
IAM::Group
IAM::ManagedPolicy
IAM::Role
IAM::User
Lambda::Function
RDS::DBCluster
RDS::DBInstance
RDS::EventSubscription
RDS::DBSecurityGroup
RDS::DBSnapshot
RDS::DBSubnetGroup
Redshift::Cluster
Redshift::ClusterParameterGroup
Redshift::ClusterSecurityGroup
Redshift::ClusterSnapshot
Redshift::ClusterSubnetGroup
S3::Bucket
SSM::ManagedInstanceInventory
SSM::Association
```

AWS AppConfig

Utilisation d'AWS AppConfig, une capacité de AWS Systems Managerpour créer, gérer et déployer rapidement des configurations d'application. à l'aide d' AppConfig prend en charge les déploiements contrôlés vers des applications de toute taille et inclut des contrôles de validation et une surveillance intégrés. Vous pouvez utiliser AWS AppConfig avec des applications hébergées sur des instances EC2, AWS Lambda, les conteneurs, les applications mobiles ou les appareils IoT.

Informations sur AWS AppConfig a été déplacé dans un guide de l'utilisateur distinct. Pour plus d'informations, consultez Qu'est-ce qu'AWS AppConfig?

AWS Systems Manager Parameter Store

AWS Systems Manager Parameter Store offre un stockage sécurisé et hiérarchique des données pour la gestion des données de configuration et des codes secrets. Vous pouvez stocker des données telles que des mots de passe, des chaînes de base de données, des Amazon Machine Image (AMI) et des codes de licence en tant que valeurs de paramètre.IDs Vous pouvez stocker ces valeurs sous forme de texte brut ou de données chiffrées. Vous pouvez référencer des paramètres Systems Manager dans vos scripts, commandes, documents SSM et flux de travail de configuration et d'automatisation à l'aide du nom unique que vous avez spécifié lors de la création du paramètre.

> Note
>
> Pour implémenter des cycles de vie de renouvellement de mot de passe, utilisez AWS Secrets Manager. Secrets Manager vous permet de renouveler, gérer et récupérer facilement des informations d'identification de base de données, des clés d'API et d'autres secrets tout au long de leur cycle de vie. Pour plus d'informations, consulter Qu'est-ce que AWS Secrets Manager ? dans le manuel AWS Secrets Manager.

Comment mon organisation peut-elle tirer parti de Parameter Store ?

Parameter Store offre les avantages suivants :

- Vous pouvez utiliser un service de gestion sécurisé, évolutif et hébergé des codes secrets, sans serveurs à gérer.
- Amélioration de vos niveaux de sécurité en séparant les données du code.
- Stockage des données de configuration et des chaînes chiffrées en hiérarchies et en versions de suivi.
- Accès de contrôle et d'audit à niveaux granulaires.

À qui est destiné Parameter Store ?

- Des clients AWS qui souhaitent disposer d'un moyen centralisé de gérer les données de configuration.
- Les développeurs de logiciels qui souhaitent stocker facilement différents ID de connexion et flux de référence.
- Des administrateurs qui souhaitent être avertis lorsque des modifications ont été apportées ou non à des secrets et des mots de passe.

Quelles sont les fonctions d'Parameter Store ?

- Notification de modification

  Vous pouvez configurer des notifications de modification et déclencher des actions automatiques pour les paramètres et les stratégies de paramètres. Pour plus d'informations, consultez la section Configuration de notifications ou d'actions de déclenchement basées sur des événements Parameter Store (p. 239).
- Organisation et contrôle de l'accès

  Vous pouvez baliser vos paramètres individuellement pour vous aider à identifier rapidement un ou plusieurs paramètres en fonction des balises que vous leur avez attribuées. Par exemple, vous pouvez baliser des paramètres pour des environnements, des services, des utilisateurs, des groupes ou des périodes spécifiques. Vous pouvez également restreindre l'accès aux paramètres en créant une stratégie IAM qui spécifie les balises auxquelles un utilisateur ou un groupe peut accéder. Pour plus d'informations, consultez la section Balisage des paramètres Systems Manager (p. 1377).
- Versions des étiquettes

  Vous pouvez associer un alias pour les versions de votre paramètre en créant des étiquettes. Les étiquettes peuvent vous aider à vous souvenir de l'objectif d'une version de paramètre lorsqu'il existe plusieurs versions.
- Validation des données

  Vous pouvez créer des paramètres qui pointent vers une Amazon EC2 instance et valident ces paramètres pour vous assurer qu'ils font référence au type de ressource attendu, que la ressource existe et que le client est autorisé à utiliser la ressource.Parameter Store Par exemple, vous pouvez créer un paramètre avec l'ID Amazon Machine Image (AMI) comme valeur avec le type de données aws:ec2:image, et Parameter Store effectuer une opération de validation asynchrone pour vous assurer que la valeur du paramètre répond aux exigences de mise en forme d'un ID d'AMI et que l'AMI spécifiée est disponible dans votre AWS compte.
- Secrets de référence

Parameter Storeest intégré à AWS Secrets Manager afin que vous puissiez récupérer les secrets Secrets Manager lorsque vous utilisez d'autres AWS services qui prennent déjà en charge les références aux Parameter Store paramètres.

- Accessible à partir d'autres AWS services

  Vous pouvez utiliser Parameter Store des paramètres avec d'autres Systems Manager fonctionnalités et AWS services pour récupérer des secrets et des données de configuration à partir d'un magasin central. Ces paramètres sont compatibles avec des fonctionnalités Systems Manager telles que Fonctionnalité Exécuter la commande, State Manager et Automation. Vous pouvez également référencer des paramètres dans un certain nombre d'autres services AWS, notamment les suivants :
  - Amazon Elastic Compute Cloud (Amazon EC2)
  - Amazon Elastic Container Service (Amazon ECS)
  - AWS Secrets Manager
  - AWS Lambda
  - AWS CloudFormation
  - AWS CodeBuild
  - AWS CodePipeline
  - AWS CodeDeploy
- Intégration à d'autres AWS services

  Configuration de l'intégration avec les services AWS suivants pour le chiffrement, la notification, la surveillance et l'audit :
  - AWS Key Management Service (AWS KMS)
  - Amazon Simple Notification Service (Amazon SNS)
  - Amazon CloudWatch: pour plus d'informations, consultez Configuration de EventBridge pour les paramètres (p. 239) .
  - Amazon EventBridge: pour plus d'informations, consultez Surveillance des changements de statut Systems Manager à l'aide de notifications Amazon SNS (p. 1344) et Référence : modèles et types Amazon EventBridge d'événements pourSystems Manager (p. 1387) .
  - AWS CloudTrail: pour plus d'informations, consultez Journalisation des appels d'API AWS Systems Manager avec AWS CloudTrail (p. 1336) .

Qu'est-ce qu'un paramètre ?

Un paramètre Parameter Store est tout élément de données enregistré dans Parameter Store, tel qu'un bloc de texte, une liste de noms, un mot de passe, un ID Amazon Machine Image (AMI), une clé de licence, etc. Vous pouvez référencer ces données de manière centralisée et sécurisée dans vos scripts, commandes et documents SSM.

Lorsque vous référencez un paramètre, vous spécifiez son nom à l'aide de la convention suivante.

{{`ssm:`*`parameter-name`*}}

> **Note**
>
> Les paramètres ne peuvent pas être référencés ou imbriqués dans les valeurs d'autres paramètres. Vous ne pouvez pas inclure `{{}}` ou `{{ssm:`*`parameter-name`*`}}` dans une valeur de paramètre.

Parameter Storeprend en charge trois types de paramètres : `String`, et `StringList` .`SecureString`

À une exception près, lorsque vous créez ou mettez à jour un paramètre, vous entrez la valeur du paramètre en texte brut et nParameter Store'effectue aucune validation sur le texte que vous

saisissez. Pour `String` les paramètres, toutefois, vous pouvez spécifier le type de données comme `aws:ec2:image` , et Parameter Store vérifie que la valeur que vous saisissez est au bon format pour une Amazon EC2 AMI ; par exemple : `ami-12345abcdeEXAMPLE`.

String (chaîne)

Par défaut, les paramètres `String` sont constitués d'un bloc de texte que vous saisissez. Exemple :

- `abc123`
- `Example Corp`
- `<img src="images/bannerImage1.png"/>`

StringList

Les paramètres `StringList` contiennent une liste de valeurs séparées par des virgules, comme le montre les exemples suivants.

```
Monday,Wednesday,Friday
```

```
CSV,TSV,CLF,ELF,JSON
```

SecureString

Un paramètre `SecureString` correspond à des données sensibles qui doivent être stockées et référencées de manière sécurisée. Si vous ne voulez pas que les utilisateurs modifient ou référencent en texte brut certaines de vos données, telles que les mots de passe ou les clés de licence, créez ces paramètres à l'aide du type de données `SecureString`.

> **Important**
>
> Ne stockez pas de données sensibles dans un paramètre `StringList` ou `String`. Pour toutes les données sensibles qui doivent rester chiffrées, utilisez uniquement le type de paramètre `SecureString`.
> Pour plus d'informations, consultez la section Créer un paramètre SecureString (AWS CLI) (p. 251).

Nous vous recommandons d'utiliser `SecureString` les paramètres pour les scénarios suivants :

- Vous voulez utiliser les données/paramètres sur les services AWS sans exposer les valeurs en texte brut dans les commandes, les fonctions, les journaux de l'agent ou les journaux AWS CloudTrail.
- Vous voulez contrôler les personnes ayant accès aux données sensibles.
- Vous souhaitez être en mesure de contrôler les accès aux données sensibles (AWS CloudTrail).
- Vous souhaitez chiffrer vos données sensibles et vous souhaitez apporter vos propres clés de chiffrement pour gérer l'accès.

> **Important**
>
> Seule la valeur d'un paramètre `SecureString` est chiffrée. Les noms de paramètre, les descriptions et les autres propriétés ne sont pas chiffrés.

Le type de `SecureString` paramètre peut être utilisé pour les données textuelles que vous souhaitez chiffrer, telles que les mots de passe, les secrets d'application, les données de configuration confidentielles ou tout autre type de données à protéger. `SecureString`Les données sont chiffrées et déchiffrées à l'aide d'une clé AWS Key Management Service (KMS). Vous pouvez utiliser une clé KMS par défaut fournie par AWS ou créer et utiliser votre propre clé principale client (CMK). (Utilisez votre propre clé CMK si vous avez besoin de restreindre l'accès des utilisateurs aux `SecureString` paramètres. Pour plus d'informations, consultez Autorisations IAM pour l'utilisation des clés par défaut AWS et des clés gérées par le client (p. 226) .)

Vous pouvez également utiliser les paramètres `SecureString` avec d'autres services AWS. Dans l'exemple suivant, la AWS Lambda fonction récupère un `SecureString` paramètre à l'aide de l'GetParametersAPI.

```
from __future__ import print_function

import json
import boto3
ssm = boto3.client('ssm', 'us-east-2')
 def get_parameters():
    response = ssm.get_parameters(
        Names=['LambdaSecureString'],WithDecryption=True
    )
    for parameter in response['Parameters']:
        return parameter['Value']

def lambda_handler(event, context):
    value = get_parameters()
    print("value1 = " + value)
    return value  # Echo back the first key value
```

> **Note**
>
> Parameter Store est également intégré à AWS Secrets Manager. Vous pouvez récupérer des Secrets Manager secrets lorsque vous utilisez d'autres AWS services qui prennent déjà en charge les références aux Parameter Store paramètres. Pour plus d'informations, consultez Référencement des secrets AWS Secrets Manager à partir des paramètres Parameter Store (p. 134) dans ce manuel.

Chiffrement et tarification AWS KMS

Si vous choisissez le type de `SecureString` paramètre lorsque vous créez votre paramètre, Systems Manager utilise AWS Key Management Service (KMS) pour chiffrer la valeur du paramètre.

Parameter Store n'applique aucun frais pour la création d'un paramètre `SecureString`, mais applique des frais pour l'utilisation du chiffrement AWS Key Management Service. Pour obtenir des informations, veuillez consulter Tarification AWS Key Management Service.

Pour plus d'informations sur AWS les gérés par et par le client, consultez CMKs Concepts AWS Key Management Service dans le AWS Key Management Service Developer Guide . Pour de plus amples informations sur Parameter Store et sur le chiffrement KMS, veuillez consulter Comment AWS Systems Manager Parameter Store utilise-t-il AWS KMS ?.

> **Note**
>
> Pour afficher une clé CMK AWS gérée par , utilisez l'AWS KMS`DescribeKey`opération. Cet AWS CLI exemple utilise `DescribeKey` pour afficher et la clé CMK gérée par .AWS

```
aws kms describe-key --key-id alias/aws/ssm
```

Voir aussi

Pour obtenir un exemple de création et d'utilisation d'un paramètre `SecureString`, veuillez consulter Créer un SecureString paramètre et joindre une instance à un domaine (PowerShell) (p. 300). Pour plus d'informations sur l'utilisation Systems Manager des paramètres avec d'autres AWS services , consultez les billets de blog suivants :

- Gestion des secrets pour les Amazon ECS applications utilisant Parameter Store et les IAM rôles pour les tâches
- Utiliser Parameter Store pour accéder en toute sécurité aux secrets et aux données de configuration dans CodeDeploy
- Articles intéressants sur Amazon EC2 Systems Manager Parameter Store

Rubriques

- Configuration d'Parameter Store (p. 223)
- Utilisation des Parameter Store (p. 242)
- Procédures Parameter Store (p. 299)
- Audit et journalisation de l'activité de Parameter Store (p. 302)
- Dépannage de Parameter Store (p. 302)

Configuration d'Parameter Store

Avant de configurer les paramètres dans Systems Manager Parameter Store , vous devez d'abord configurer les stratégies AWS Identity and Access Management (IAM) qui accordent aux utilisateurs de votre compte l'autorisation d'effectuer les actions que vous spécifiez. Cette section inclut des informations sur la façon de configurer manuellement ces stratégies à l'aide de la console IAM, et la façon de les affecter à des utilisateurs et groupes d'utilisateurs. Vous pouvez également créer et attribuer des stratégies pour contrôler les actions de paramètres qui peuvent être exécutées sur une instance. Cette section inclut aussi des informations sur la manière de créer des règles Amazon EventBridge qui vous permettent de recevoir des notifications sur les modifications apportées aux paramètres Systems Manager. Vous pouvez également utiliser des règles EventBridge pour déclencher d'autres actions dans AWS fonction des modifications dans Parameter Store.

Table des matières

- Restriction de l'accès aux paramètres Systems Manager à l'aide des stratégies IAM (p. 223)
- Gestion des niveaux de paramètres (p. 228)
- Accroissement du débit Parameter Store (p. 235)
- Configuration de notifications ou d'actions de déclenchement basées sur des événements Parameter Store (p. 239)

Restriction de l'accès aux paramètres Systems Manager à l'aide des stratégies IAM

Vous limitez l'accès aux paramètres Systems Manager en utilisant AWS Identity and Access Management (IAM). Plus précisément, vous créez des stratégies IAM qui restreignent l'accès aux opérations API suivantes :

- DeleteParameter
- DeleteParameters
- DescribeParameters
- GetParameter
- GetParameters
- GetParameterHistory
- GetParametersByPath
- PutParameter

Lorsque vous utilisez IAM des stratégies pour restreindre l'accès aux Systems Manager paramètres, nous vous recommandons de créer et d'utiliser des stratégies IAM restrictives. Par exemple, la stratégie suivante permet à un utilisateur d'appeler les opérations d'API `DescribeParameters` et `GetParameters` pour un ensemble limité de ressources. Cela signifie que l'utilisateur peut obtenir des informations sur les paramètres qui commencent par `prod-*` et les utiliser.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeParameters"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetParameters"
            ],
            "Resource": "arn:aws:ssm:us-east-2:123456789012:parameter/prod-*"
        }
    ]
}
```

Pour les administrateurs de confiance, vous pouvez fournir un accès complet à toutes les opérations d'API des paramètres Systems Manager en utilisant une stratégie similaire à l'exemple suivant. Cette stratégie accorde à l'utilisateur un accès complet à tous les paramètres de production qui commencent par `dbserver-prod-*`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "ssm:PutParameter",
                "ssm:DeleteParameter",
                "ssm:GetParameterHistory",
                "ssm:GetParametersByPath",
                "ssm:GetParameters",
                "ssm:GetParameter",
                "ssm:DeleteParameters"
            ],
            "Resource": "arn:aws:ssm:region:account-id:parameter/dbserver-prod-*"
        },
        {
            "Sid": "VisualEditor1",
            "Effect": "Allow",
            "Action": "ssm:DescribeParameters",
            "Resource": "*"
        }
    ]
}
```

Rubriques

- Refuser des autorisations (p. 224)
- Autoriser uniquement à des paramètres spécifiques d'être exécutée sur des instances (p. 226)
- Autorisations IAM pour l'utilisation des clés par défaut AWS et des clés gérées par le client (p. 226)

Refuser des autorisations

Chaque API est unique et dispose d'opérations et d'autorisations distinctes que vous pouvez accorder ou refuser individuellement. Un refus explicite dans une stratégie remplace l'autorisation.

Note

La clé (AWS Key Management Service) par défaut est AWS KMS autorisée pour tous les `Decrypt` mandataires du IAM compte.AWS Si vous souhaitez avoir différents niveaux d'accès aux `SecureString` paramètres de votre compte, nous vous déconseillons d'utiliser la clé par défaut.

Si vous voulez que toutes les opérations d'API récupérant les valeurs des paramètres aient le même comportement, vous pouvez utiliser un modèle comme `GetParameter*` dans une stratégie. L'exemple suivant montre comment refuser `GetParameter` , `GetParameters` , et `GetParameterHistory` pour tous les paramètres commençant par `GetParametersByPath` .`prod-*`

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "ssm:GetParameter*"
            ],
            "Resource": "arn:aws:ssm:us-east-2:123456789012:parameter/prod-*"
        }
    ]
}
```

L'exemple suivant montre comment refuser certaines commandes tout en autorisant l'utilisateur à exécuter d'autres commandes sur tous les paramètres commençant par `prod-*` .

```
{
    "Version": "2012-10-17",
    "Statement": [
            "Effect": "Deny",
            "Action": [
                "ssm:PutParameter",
                "ssm:DeleteParameter",
                "ssm:DeleteParameters"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetParametersByPath",
                "ssm:GetParameters",
                "ssm:GetParameter",
                "ssm:GetParameterHistory"
                "ssm:DescribeParameters"
            ],
            "Resource": "arn:aws:ssm:region:account-id:parameter/prod-*"
        }
    ]
}
```

Note

L'historique des paramètres inclut toutes les versions des paramètres, y compris la version actuelle. Par conséquent, si un utilisateur se voit refuser l'autorisation pour `GetParameter` , et `GetParameters` , `GetParameterByPath` mais reçoit l'autorisation pour , il peut voir le paramètre actuel, y compris `GetParameterHistory` les paramètres, à l'aide de `SecureString` .`GetParameterHistory`

Autoriser uniquement à des paramètres spécifiques d'être exécutée sur des instances

Vous pouvez contrôler l'accès afin que les instances puissent uniquement exécuter les paramètres que vous spécifiez.

Si vous choisissez le type de `SecureString` paramètre lorsque vous créez votre paramètre, Systems Manager utilise AWS Key Management Service (AWS KMS) pour chiffrer la valeur du paramètre. chiffre la valeur à l'aide d'une clé principale client gérée AWS KMS par (CMK) ou d'une clé CMK gérée par le client.AWS Pour plus d'informations sur AWS KMS et CMKs, consultez le AWS Key Management Service Developer Guide.

Vous pouvez afficher la clé CMK AWS gérée par en exécutant la commande suivante à partir de l' AWS CLI .

```
aws kms describe-key --key-id alias/aws/ssm
```

L'exemple suivant permet aux instances d'obtenir une valeur de paramètre uniquement pour les paramètres commençant par `prod-` . Si le paramètre est un `SecureString` paramètre, l'instance déchiffre la chaîne à l'aide d' AWS KMS .

> **Note**
>
> Les stratégies d'instances, comme dans l'exemple précédent, sont attribuées au rôle de l'instance dans IAM. Pour plus d'informations sur la configuration de l'accès aux fonctions Systems Manager, y compris la façon d'attribuer des stratégies aux utilisateurs et aux instances, consultez Configuration d'AWS Systems Manager (p. 28).

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "ssm:GetParameters"
         ],
         "Resource":[
            "arn:aws:ssm:region:account-id:parameter/prod-*"
         ]
      },
      {
         "Effect":"Allow",
         "Action":[
            "kms:Decrypt"
         ],
         "Resource":[
            "arn:aws:kms:region:account-id:key/CMK"
         ]
      }
   ]
}
```

Autorisations IAM pour l'utilisation des clés par défaut AWS et des clés gérées par le client

Les paramètres Parameter Store `SecureString` sont chiffrés et déchiffrés à l'aide de clés AWS Key Management Service (AWS KMS). Vous pouvez choisir de chiffrer vos `SecureString` paramètres à l'aide d'une clé principale client (CMK) ou de la clé KMS par défaut fournie par AWS .

Lorsque vous utilisez une clé gérée par le client, la IAM stratégie qui accorde à un utilisateur l'accès à un paramètre ou à un chemin d'accès doit fournir `kms:Encrypt` des autorisations explicites pour la clé. Par

exemple, la stratégie suivante permet à un utilisateur de créer, mettre à jour et afficher les `SecureString` paramètres qui commencent par `prod-` dans la AWS région et le compte spécifiés.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:PutParameter",
                "ssm:GetParameter",
                "ssm:GetParameters"
            ],
            "Resource": [
                "arn:aws:ssm:us-east-2:111122223333:parameter/prod-*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
               "kms:Decrypt",
               "kms:Encrypt",
               "kms:GenerateDataKey"  1
            ],
            "Resource": [
                "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-12345EXAMPLE"
            ]
        }
    ]
}
```

[1]L'autorisation `kms:GenerateDataKey` est requise pour créer des paramètres avancés chiffrés à l'aide de la clé gérée par le client spécifiée.

En revanche, tous les utilisateurs du compte client ont accès à la clé AWS gérée par défaut. Si vous utilisez cette clé par défaut pour chiffrer `SecureString` des paramètres et que vous ne souhaitez pas que les utilisateurs utilisent `SecureString` ces paramètres, leurs IAM stratégies doivent explicitement refuser l'accès à la clé par défaut, comme illustré dans l'exemple de stratégie suivant.

> **Note**
>
> Vous pouvez localiser l'Amazon Resource Name (ARN) de la clé par défaut dans la AWS KMS console sur la page des clés gérées par AWS. La clé par défaut est celle identifiée par `aws/ssm` dans la colonne Alias.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "kms:Decrypt",
                "kms:GenerateDataKey "
            ],
            "Resource": [
                "default-key-ARN"
            ]
        }
    ]
}
```

Si vous avez besoin d'un contrôle d'accès précis sur les paramètres `SecureString` de votre compte, vous devez utiliser une clé CMK gérée par le client pour protéger et restreindre l'accès à ces paramètres. Nous vous recommandons également d'utiliser AWS CloudTrail pour surveiller les activités des paramètres `SecureString`.

Pour de plus amples informations, consultez les rubriques suivantes :

- Logique d'évaluation de stratégies dans le IAM Guide de l'utilisateur
- Utilisation de stratégies de clé dans AWS KMS dans le AWS Key Management Service Developer Guide
- Affichage des événements avec l'historique des événements CloudTrail dans le AWS CloudTrail User Guide

Gestion des niveaux de paramètres

AWS Systems Manager Parameter Store inclut des paramètres standard et des paramètres avancés. Vous configurez individuellement les paramètres afin qu'ils utilisent le niveau paramètre standard (niveau par défaut) ou le niveau paramètre avancé.

Vous pouvez remplacer un paramètre standard par un paramètre avancé à tout moment, mais vous ne pouvez pas remplacer un paramètre avancé par un paramètre standard. En effet, le remplacement d'un paramètre avancé par un paramètre standard entraînerait une perte de données, car le système réduirait de 8 Ko à 4 Ko la taille du paramètre. Cela entraînerait également la suppression de toutes les stratégies associées au paramètre. Par ailleurs, les paramètres avancés utilisent une forme de chiffrement différente de celle des paramètres standard. Pour plus d'informations, consultez Comment AWS Systems Manager Parameter Store utilise-t-il AWS KMS ? dans le AWS Key Management Service Developer Guide.

Si vous n'avez plus besoin d'un paramètre avancé ou si vous ne souhaitez plus payer de frais supplémentaires pour un paramètre avancé, vous devez le supprimer et le recréer en tant que nouveau paramètre standard.

Le tableau suivant décrit les différences entre les niveaux.

	Standard	Advanced (Avancé)
Nombre total de paramètres autorisés (par région et compte AWS)	10 000	100 000
Taille maximale d'une valeur de paramètre.	4 Ko	8 Ko
Stratégies de paramètre disponibles	Non	Oui Pour plus d'informations, consultez la section Affectation de stratégies de paramètres (p. 273).
Coût	Pas de frais supplémentaires	Des frais supplémentaires seront facturés Pour de plus amples informations, veuillez consulter Tarification AWS Systems Manager.

Rubriques

- Spécification d'un niveau de paramètre par défaut (p. 229)
- Remplacement d'un paramètre standard par un paramètre avancé (p. 234)

Spécification d'un niveau de paramètre par défaut

Dans les demandes de création ou de mise à jour d'un paramètre (c'est-à-dire, l'action `PutParameter`), vous pouvez spécifier le niveau de paramètre à utiliser dans la demande. Voici un exemple avec utilisation de l'AWS CLI.

```
aws ssm put-parameter \
    --name "default-ami" \
    --type "String" \
    --value "t2.micro" \
    --tier "Standard"
```

Chaque fois que vous spécifiez un niveau dans la demande, Parameter Store crée ou met à jour le paramètre en fonction de votre demande. Toutefois, si vous ne spécifiez pas explicitement un niveau dans une demande, c'est le paramètre de niveau par défaut de Parameter Store qui détermine dans quel niveau le paramètre est créé.

Le niveau par défaut lorsque vous commencez à utiliser Parameter Store est le niveau de paramètre standard. Si vous utilisez le niveau de paramètre avancé, vous pouvez spécifier l'une des valeurs suivantes par défaut :

- Avancé : Avec cette option, Parameter Store évalue toutes les demandes en tant que paramètres avancés.
- Intelligent-Tiering (Hiérarchisation intelligente) : Avec cette option, Parameter Store évalue chaque demande pour déterminer si le paramètre est standard ou avancé.

  Si la demande n'inclut aucune option nécessitant un paramètre avancé, le paramètre est créé dans le niveau paramètre standard. Si une ou plusieurs options nécessitant un paramètre avancé sont incluses dans la demande, Parameter Store crée un paramètre dans le niveau paramètre avancé.

Avantages de la fonction Intelligent-Tiering (Hiérarchisation intelligente)

Voici les raisons pouvant justifier le choix de la fonction Intelligent-Tiering (Hiérarchisation intelligente) comme niveau par défaut.

Contrôle des coûts - La fonction Intelligent-Tiering permet de contrôler les coûts liés aux paramètres en créant toujours des paramètres standard, sauf si un paramètre avancé est absolument nécessaire.

Mise à niveau automatique vers le niveau paramètre avancé - Lorsque vous apportez à votre code une modification nécessitant la mise à niveau d'un paramètre standard vers un paramètre avancé, la fonction Intelligent-Tiering gère la conversion pour vous. Vous n'avez pas besoin de modifier votre code pour gérer la mise à niveau.

Voici quelques exemples de mise à niveau automatique :

- Vos modèles AWS CloudFormation fournissent de nombreux paramètres lorsqu'ils sont exécutés. Lorsque ce processus vous conduit à atteindre la limite de 10 000 paramètres dans le niveau paramètre standard, la fonction Intelligent-Tiering vous met automatiquement à niveau vers le niveau de paramètre avancé et vos processus AWS CloudFormation ne sont pas interrompus.
- Vous stockez une valeur de certificat dans un paramètre, effectuez une rotation régulière de la valeur de certificat et le contenu est inférieur à la limite de 4 Ko du niveau paramètre standard. Si une valeur de certificat de remplacement dépasse 4 Ko, la fonction Intelligent-Tiering met automatiquement à niveau le paramètre vers le niveau paramètre avancé.

- Vous souhaitez associer de nombreux paramètres standard existants à une stratégie de paramètre, ce qui nécessite le niveau paramètre avancé. Au lieu d'inclure l'option `--tier Advanced` dans tous les appels de mise à à jour les paramètres, la fonction Intelligent-Tiering met automatiquement à niveau les paramètres vers le niveau paramètre avancé. La fonction Intelligent-Tiering effectue une mise à niveau des paramètres du statut standard au statut avancé chaque fois que des critères pour le niveau paramètre avancé sont ajoutés.

Les options qui nécessitent un paramètre avancé sont les suivantes :

- La taille du contenu du paramètre est supérieure à 4 Ko.
- Le paramètre utilise une stratégie de paramètre.
- Plus de 10 000 paramètres existent déjà dans votre compte AWS dans la région actuelle.

Options de niveau par défaut

Les options de niveau que vous pouvez spécifier comme options par défaut sont les suivantes.

- Standard– Le niveau paramètre standard est le niveau par défaut lorsque vous commencez à utiliser Parameter Store. En utilisant le niveau paramètre standard, vous pouvez créer 10 000 paramètres pour chaque région dans un compte AWS. La taille du contenu de chaque paramètre peut être égale à un maximum de 4 Ko. Les paramètres standard ne prennent pas en charge les stratégies de paramètre. L'utilisation du niveau paramètre standard n'entraîne pas de frais supplémentaires. Si vous choisissez Standard comme niveau par défaut, Parameter Store tente en permanence de créer un paramètre standard pour les demandes qui ne spécifient pas de niveau.
- Avancé – Le niveau paramètre avancé vous permet de créer un maximum de 100 000 paramètres pour chaque région dans un compte AWS. La taille du contenu de chaque paramètre peut être égale à un maximum de 8 Ko. Les paramètres avancés prennent en charge les stratégies de paramètre. L'utilisation du niveau paramètre avancé est facturée. Pour de plus amples informations, veuillez consulter Tarification AWS Systems Manager. Si vous choisissez Avancé comme niveau par défaut, Parameter Store tente en permanence de créer un paramètre avancé pour les demandes qui ne spécifient pas de niveau.

  Note

  Lorsque vous choisissez le niveau paramètre avancé, vous devez autoriser explicitement AWS à facturer à votre compte pour tous les paramètres avancés que vous créez.
- Intelligent-Tiering (Hiérarchisation intelligente) – L'option Intelligent-Tiering permet à Parameter Store de déterminer s'il convient d'utiliser le niveau paramètre standard ou paramètre avancé en fonction du contenu de la demande. Par exemple, si vous exécutez une commande pour créer un paramètre avec un contenu inférieur à 4 Ko et qu'il y a moins de 10 000 paramètres dans la région actuelle de votre compte AWS, si vous ne spécifiez pas de stratégie de paramètre, un paramètre standard est créé. Si vous exécutez une commande pour créer un paramètre avec plus de 4 Ko de contenu, que vous avez déjà plus de 10 000 paramètres dans la région actuelle de votre compte AWS ou que vous spécifiez une stratégie de paramètre, un paramètre avancé est créé.

  Note

  Lorsque vous choisissez Intelligent-Tiering, vous devez autoriser explicitement AWS à facturer votre compte pour tous les paramètres avancés créés.

Vous pouvez modifier le paramètre de niveau Parameter Store par défaut à tout moment.

Configuration des autorisations de spécification d'un niveau Parameter Store par défaut

Vérifiez que vous avez l'autorisation, dans AWS Identity and Access Management (IAM), de modifier le niveau de paramètre par défaut dans Parameter Store en effectuant l'une des actions suivantes :

- Assurez-vous que la stratégie `AdministratorAccess` est attachée à votre utilisateur, groupe ou rôle IAM.
- Assurez-vous que vous avez l'autorisation de modifier le paramètre de niveau par défaut à l'aide des actions d'API suivantes :
  - GetServiceSetting
  - UpdateServiceSetting
  - ResetServiceSetting

Utilisez la procédure suivante pour ajouter une stratégie IAM en ligne à un compte utilisateur. Cette stratégie permet à un utilisateur d'afficher et de modifier le paramètre de niveau par défaut pour les paramètres d'une région spécifique dans un compte AWS.

1. Connectez-vous à la AWS Management Console et ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation, sélectionnez Users (Utilisateurs).
3. Dans la liste, sélectionnez le nom de l'utilisateur auquel attacher une stratégie.
4. Choisissez l'onglet Autorisations.
5. Sur le côté droit de la page, sous Permission policies (Stratégies d'autorisation), choisissez Add inline policy (Ajouter une stratégie en ligne).
6. Choisissez l'onglet JSON.
7. Remplacez le contenu par défaut par ce qui suit :

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetServiceSetting"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:UpdateServiceSetting"
            ],
            "Resource": "arn:aws:ssm:region:account-id:servicesetting/ssm/parameter-
store/default-parameter-tier"
        }
    ]
}
```

8. Choisissez Examiner une stratégie.
9. Sur la page Examiner une stratégie, pour Nom, entrez un nom pour la stratégie en ligne, par exemple **`Parameter-Store-Default-Tier`** ou un autre nom de votre choix.
10. Choisissez Create policy (Créer une stratégie).

Les administrateurs peuvent spécifier une autorisation en lecture seule en affectant la stratégie en ligne suivante au compte de l'utilisateur.

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

```
            "Effect": "Allow",
            "Action": [
                "ssm:GetServiceSetting"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Deny",
            "Action": [
                "ssm:ResetServiceSetting",
                "ssm:UpdateServiceSetting"
            ],
            "Resource": "*"
        }
    ]
}
```

Pour de plus amples informations sur la création et la modification de stratégies IAM, veuillez consulter Création de stratégies IAM dans le IAM Guide de l'utilisateur.

Spécification ou modification du niveau Parameter Store par défaut (console)

La procédure suivante montre comment utiliser la console Systems Manager pour spécifier ou modifier le niveau de paramètre par défaut pour le compte et la région AWS actuels.

> **Tip**
>
> Si vous n'avez pas encore créé de paramètre, vous pouvez utiliser l'AWS CLI ou Outils pour Windows PowerShell pour modifier le niveau de paramètre par défaut. Pour plus d'informations, consultez Spécification ou modification du niveau Parameter Store par défaut (AWS CLI) (p. 233) et Spécification ou modification du niveau Parameter Store par défaut (PowerShell) (p. 233).

Pour spécifier ou modifier le niveau Parameter Store par défaut

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Parameter Store.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Parameter Store.
3. Choisissez l'onglet Settings (Paramètres).
4. Choisissez Modifier le niveau par défaut.
5. Choisissez l'une des options suivantes :

   - Standard
   - Advanced (Avancé)
   - Intelligent-Tiering (Hiérarchisation intelligente)

   Pour de plus amples informations sur ces options, veuillez consulter Spécification d'un niveau de paramètre par défaut (p. 229).
6. Examinez le message, puis choisissez Confirm (Confirmer).

Si vous souhaitez modifier le paramètre de niveau par défaut ultérieurement, répétez cette procédure et spécifiez une autre option de niveau par défaut.

Spécification ou modification du niveau Parameter Store par défaut (AWS CLI)

La procédure suivante montre comment utiliser l'AWS Command Line Interface (AWS CLI) pour modifier le paramètre de niveau par défaut pour le compte et la région AWS actuels.

Pour spécifier ou modifier le niveau Parameter Store par défaut à l'aide de l'AWS CLI

1. Ouvrez l'AWS CLI et exécutez la commande suivante pour modifier le paramètre de niveau par défaut pour une région spécifique dans un compte AWS.

   ```
   aws ssm update-service-setting --setting-id arn:aws:ssm:region:account-
   id:servicesetting/ssm/parameter-store/default-parameter-tier --setting-value tier-
   option
   ```

   région représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de *region* prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

   tier-option Les valeurs incluent `Standard` , et `Advanced` .`Intelligent-Tiering` Pour plus d'informations sur ces options, consultez Spécification d'un niveau de paramètre par défaut (p. 229) .

   Il n'y a pas de sortie si la commande réussit.

2. Exécutez la commande suivante pour afficher les paramètres de service de débit actuels pour Parameter Store dans le compte et la région AWS actuels.

   ```
   aws ssm get-service-setting --setting-id arn:aws:ssm:region:account-id:servicesetting/
   ssm/parameter-store/default-parameter-tier
   ```

   Le système renvoie des informations similaires à ce qui suit :

   ```
   {
       "ServiceSetting": {
           "SettingId": "/ssm/parameter-store/default-parameter-tier",
           "SettingValue": "Advanced",
           "LastModifiedDate": 1556551683.923,
           "LastModifiedUser": "arn:aws:sts::123456789012:assumed-role/Administrator/
   Jasper",
           "ARN": "arn:aws:ssm:us-east-2:123456789012:servicesetting/ssm/parameter-store/
   default-parameter-tier",
           "Status": "Customized"
       }
   }
   ```

Si vous souhaitez modifier à nouveau le paramètre de niveau par défaut, répétez cette procédure et spécifiez une autre option `SettingValue`.

Spécification ou modification du niveau Parameter Store par défaut (PowerShell)

La procédure suivante montre comment utiliser Outils AWS pour Windows PowerShell pour modifier le paramètre de niveau par défaut pour une région spécifique dans un compte AWS.

Pour spécifier ou modifier le niveau Parameter Store par défaut à l'aide dePowerShell

1. Modifiez le niveau Parameter Store par défaut dans le compte et la région AWS actuels à l'aide d'Outils AWS pour PowerShell.

```
Update-SSMServiceSetting -SettingId "arn:aws:ssm:region:account-id:servicesetting/ssm/
parameter-store/default-parameter-tier" -SettingValue "tier-option" -Region region
```

région représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de *region* prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

tier-option Les valeurs incluent `Standard` , et `Advanced` .`Intelligent-Tiering` Pour plus d'informations sur ces options, consultez Spécification d'un niveau de paramètre par défaut (p. 229) .

Il n'y a pas de sortie si la commande réussit.

2. Exécutez la commande suivante pour afficher les paramètres de service de débit actuels pour Parameter Store dans le compte et la région AWS actuels.

```
Get-SSMServiceSetting -SettingId "arn:aws:ssm:region:account-id:servicesetting/ssm/
parameter-store/default-parameter-tier" -Region region
```

région représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de *region* prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

Le système renvoie des informations similaires à ce qui suit :

```
ARN              : arn:aws:ssm:us-east-2:123456789012:servicesetting/ssm/parameter-
store/default-parameter-tier
LastModifiedDate : 4/29/2019 3:35:44 PM
LastModifiedUser : arn:aws:sts::123456789012:assumed-role/Administrator/Jasper
SettingId        : /ssm/parameter-store/default-parameter-tier
SettingValue     : Advanced
Status           : Customized
```

Si vous souhaitez modifier à nouveau le paramètre de niveau par défaut, répétez cette procédure et spécifiez une autre option `SettingValue`.

Remplacement d'un paramètre standard par un paramètre avancé

Utilisez la procédure suivante pour remplacer un paramètre standard existant par un paramètre avancé. Pour de plus amples informations sur la création d'un nouveau paramètre avancé, veuillez consulter Création de paramètres Systems Manager (p. 243).

Pour remplacer un paramètre standard par un paramètre avancé

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Parameter Store (Stockage de paramètres).
3. Choisissez un paramètre, puis choisissez Edit (Modifier).
4. Pour Description, entrez les informations concernant ce paramètre.
5. Sélectionnez Advanced.
6. Pour Value (Valeur), saisissez la valeur de ce paramètre. La valeur maximale des paramètres avancés est de 8 Ko.
7. Sélectionnez Save Changes (Enregistrer les modifications).

Accroissement du débit Parameter Store

L'augmentation du débit de Parameter Store permet d'augmenter le nombre maximal de transactions par seconde (TPS) pouvant être traitées par Parameter Store. L'augmentation du débit vous permet d'exploiter Parameter Store à des volumes plus élevés afin de prendre en charge des applications et des charges de travail nécessitant des accès simultanés à un grand nombre de paramètres. Vous pouvez faire passer la limite à 1 000 TPS dans l'onglet Settings (Paramètres). L'augmentation de la limite du débit entraîne la facturation de frais supplémentaires sur votre compte AWS. Pour de plus amples informations, veuillez consulter Tarification AWS Systems Manager.

> **Note**
>
> Le paramètre de débit de Parameter Store s'appliquent à toutes les transactions créées par tous les utilisateurs AWS Identity and Access Management (IAM) dans le compte et la région AWS actuels. Le paramètre de débit s'applique aux paramètres standard et avancés.

Rubriques

- Configuration des autorisations d'augmentation du débit de Parameter Store (p. 235)
- Augmentation du débit (console) (p. 236)
- Accroissement du débit (AWS CLI) (p. 237)
- Augmentation du débit (PowerShell) (p. 238)

Configuration des autorisations d'augmentation du débit de Parameter Store

Vérifiez que vous êtes autorisé dans AWS Identity and Access Management (IAM) à augmenter le débit de Parameter Store en effectuant l'une des actions suivantes :

- Assurez-vous que la stratégie `AdministratorAccess` est attachée à votre utilisateur, groupe ou rôle IAM.
- Assurez-vous que vous avez l'autorisation de modifier le paramètre de service de débit à l'aide des actions d'API suivantes :
  - GetServiceSetting
  - UpdateServiceSetting
  - ResetServiceSetting

Utilisez la procédure suivante pour ajouter une stratégie IAM en ligne à un compte utilisateur. Cette stratégie permet à un utilisateur d'afficher et de modifier le paramètre parameter-throughput pour les paramètres associés à leur compte et leur région.

1. Connectez-vous à la AWS Management Console et ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation, sélectionnez Users (Utilisateurs).
3. Dans la liste, sélectionnez le nom de l'utilisateur auquel attacher une stratégie.
4. Choisissez l'onglet Autorisations.
5. Sur le côté droit de la page, sous Permission policies (Stratégies d'autorisation), choisissez Add inline policy (Ajouter une stratégie en ligne).
6. Choisissez l'onglet JSON.
7. Remplacez le contenu par défaut par ce qui suit :

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

```
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetServiceSetting"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:UpdateServiceSetting"
            ],
            "Resource": "arn:aws:ssm:region:account-id:servicesetting/ssm/parameter-
store/high-throughput-enabled"
        }
    ]
}
```

8. Choisissez Examiner une stratégie.
9. Sur la page Examiner une stratégie, pour Nom, entrez un nom pour la stratégie en ligne, par exemple **`Parameter-Store-Throughput`** ou un autre nom de votre choix.
10. Choisissez Create policy (Créer une stratégie).

Les administrateurs peuvent spécifier une autorisation en lecture seule en affectant la stratégie en ligne suivante au compte de l'utilisateur.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetServiceSetting"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Deny",
            "Action": [
                "ssm:ResetServiceSetting",
                "ssm:UpdateServiceSetting"
            ],
            "Resource": "*"
        }
    ]
}
```

Pour de plus amples informations sur la création et la modification de stratégies IAM, veuillez consulter Création de stratégies IAM dans le IAM Guide de l'utilisateur.

Augmentation du débit (console)

La procédure suivante montre comment utiliser la console Systems Manager pour augmenter le nombre de transactions par seconde pouvant être traitées par Parameter Store pour le compte et la région AWS actuels.

> **Tip**
>
> Si vous n'avez pas encore créé de paramètre, vous pouvez utiliser l'AWS CLI ou Outils AWS pour Windows PowerShell pour accroître le débit. Pour plus d'informations, consultez Accroissement du débit (AWS CLI) (p. 237) et Augmentation du débit (PowerShell) (p. 238).

Pour augmenter le débit Parameter Store

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Parameter Store.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Parameter Store.
3. Choisissez l'onglet Settings (Paramètres).
4. Choisissez Set limit (Définir la limite).
5. Examinez le message, puis choisissez Accept (Accepter).

Si vous n'avez plus besoin d'un débit plus élevé ou si vous ne souhaitez plus payer de frais supplémentaires, vous pouvez rétablir les paramètres standard. Pour rétablir vos paramètres, répétez cette procédure, puis choisissez Reset limit (Réinitialiser la limite).

Accroissement du débit (AWS CLI)

La procédure suivante montre comment utiliser l'AWS Command Line Interface (AWS CLI) pour augmenter le nombre de transactions par seconde pouvant être traitées par Parameter Store pour le compte et la région AWS actuels.

Pour augmenter le débit Parameter Store à l'aide de l'AWS CLI

1. Ouvrez l'AWS CLI et exécutez la commande suivante pour augmenter le nombre de transactions par seconde pouvant être traitées par Parameter Store dans le compte et la région AWS actuels.

   ```
   aws ssm update-service-setting --setting-id arn:aws:ssm:region:account-
   id:servicesetting/ssm/parameter-store/high-throughput-enabled --setting-value true
   ```

   Il n'y a pas de sortie si la commande réussit.
2. Exécutez la commande suivante pour afficher les paramètres de service de débit actuels pour Parameter Store dans le compte et la région AWS actuels.

   ```
   aws ssm get-service-setting --setting-id arn:aws:ssm:region:account-id:servicesetting/
   ssm/parameter-store/high-throughput-enabled
   ```

   ```
   {
       "ServiceSetting": {
           "SettingId": "/ssm/parameter-store/high-throughput-enabled",
           "SettingValue": "true",
           "LastModifiedDate": 1556551683.923,
           "LastModifiedUser": "arn:aws:sts::123456789012:assumed-role/Administrator/
   Jasper",
           "ARN": "arn:aws:ssm:us-east-2:123456789012:servicesetting/ssm/parameter-store/
   high-throughput-enabled",
           "Status": "Customized"
       }
   }
   ```

Si vous n'avez plus besoin d'un débit plus élevé ou si vous ne souhaitez plus payer de frais supplémentaires, vous pouvez rétablir les paramètres standard. Pour réinitialiser vos paramètres, exécutez la commande suivante.

```
aws ssm reset-service-setting --setting-id arn:aws:ssm:region:account-id:servicesetting/
ssm/parameter-store/high-throughput-enabled
```

```
{
    "ServiceSetting": {
        "SettingId": "/ssm/parameter-store/high-throughput-enabled",
        "SettingValue": "false",
        "LastModifiedDate": 1555532818.578,
        "LastModifiedUser": "System",
        "ARN": "arn:aws:ssm:us-east-2:123456789012:servicesetting/ssm/parameter-store/high-
throughput-enabled",
        "Status": "Default"
    }
}
```

Augmentation du débit (PowerShell)

La procédure suivante vous montre comment utiliser les outils Outils AWS pour Windows PowerShell pour augmenter le nombre de transactions par seconde pouvant être traitées par Parameter Store pour le compte et la région AWS actuels.

Pour augmenter Parameter Store le débit à l'aide dePowerShell

1. Augmentez le débit de Parameter Store dans le compte et la région AWS actuels à l'aide des Outils AWS pour PowerShell.

   ```
   Update-SSMServiceSetting -SettingId "arn:aws:ssm:region:account-id:servicesetting/ssm/
   parameter-store/high-throughput-enabled" -SettingValue "true" -Region us-east-2
   ```

   Il n'y a pas de sortie si la commande réussit.
2. Exécutez la commande suivante pour afficher les paramètres de service de débit actuels pour Parameter Store dans le compte et la région AWS actuels.

   ```
   Get-SSMServiceSetting -SettingId "arn:aws:ssm:region:account-id:servicesetting/ssm/
   parameter-store/high-throughput-enabled" -Region us-east-2
   ```

   ```
   ARN              : arn:aws:ssm:us-east-2:123456789012:servicesetting/ssm/parameter-
   store/high-throughput-enabled
   LastModifiedDate : 4/29/2019 3:35:44 PM
   LastModifiedUser : arn:aws:sts::123456789012:assumed-role/Administrator/Jasper
   SettingId        : /ssm/parameter-store/high-throughput-enabled
   SettingValue     : true
   Status           : Customized
   ```

Si vous n'avez plus besoin d'un débit plus élevé ou si vous ne souhaitez plus payer de frais supplémentaires, vous pouvez rétablir les paramètres standard. Pour réinitialiser vos paramètres, exécutez la commande suivante.

```
Reset-SSMServiceSetting -SettingId "arn:aws:ssm:region:account-id:servicesetting/ssm/
parameter-store/high-throughput-enabled" -Region region
```

```
ARN              : arn:aws:ssm:us-east-2:123456789012:servicesetting/ssm/parameter-store/
high-throughput-enabled
LastModifiedDate : 4/17/2019 8:26:58 PM
LastModifiedUser : System
```

```
SettingId        : /ssm/parameter-store/high-throughput-enabled
SettingValue     : false
Status           : Default
```

Configuration de notifications ou d'actions de déclenchement basées sur des événements Parameter Store

Les rubriques de cette section expliquent comment utiliser Amazon EventBridge et Amazon Simple Notification Service (Amazon SNS) pour que les changements au niveau des paramètres Systems Manager vous soient indiqués. Vous pouvez créer une EventBridge règle pour être averti lorsqu'un paramètre ou une version d'étiquette de paramètre est créé, mis à jour ou supprimé. Vous pouvez être informé des changements ou du statut lié aux stratégies de paramètre, par exemple lorsqu'un paramètre expire, va expirer ou n'a pas changé pendant un laps de temps spécifié.

> **Note**
>
> Les stratégies de paramètre sont disponibles pour les paramètres qui utilisent le niveau de paramètres avancés. Des frais supplémentaires seront facturés. Pour de plus amples informations, veuillez consulter Affectation de stratégies de paramètres (p. 273) et Gestion des niveaux de paramètres (p. 228).

Les rubriques ci-dessous expliquent également comment déclencher d'autres actions sur une cible pour des événements de paramètres spécifiques. Par exemple, vous pouvez exécuter une fonction AWS Lambda pour recréer automatiquement un paramètre lorsqu'il arrive à expiration ou est supprimé. Vous pouvez configurer une notification pour déclencher une fonction Lambda lorsque le mot de passe de votre base de données est mis à jour. La fonction Lambda peut forcer vos connexions de base de données à se réinitialiser ou à se reconnecter avec le nouveau mot de passe. EventBridge prend également en charge l'exécution de commandes Fonctionnalité Exécuter la commande et d'exécutions d'automatisations, et d'actions dans de nombreux autres services AWS. Pour de plus amples informations, veuillez consulter Guide de l'utilisateur Amazon EventBridge.

Avant de commencer

Créez toutes les ressources dont vous avez besoin pour spécifier l'action cible pour la règle que vous créez. Par exemple, si la règle que vous créez concerne l'envoi d'une notification, créez d'abord une rubrique Amazon SNS. Pour de plus amples informations, veuillez consulter Mise en route avec Amazon SNS dans le Amazon Simple Notification Service Manuel du développeur.

Rubriques

- Configuration de EventBridge pour les paramètres (p. 239)
- Configuration de EventBridge pour les stratégies de paramètres (p. 241)

Configuration de EventBridge pour les paramètres

Cette rubrique explique comment créer une EventBridge règle qui appelle une cible en fonction des événements qui se produisent sur un ou plusieurs paramètres de votre AWS compte.

Pour configurer EventBridge pour les paramètres Systems Manager

1. Ouvrez la console Amazon EventBridge à l'adresse https://console.aws.amazon.com/events/.
2. Dans le volet de navigation, choisissez Rules (Règles), puis Create rule (Créer une règle).

   -ou-

   Si la page Amazon EventBridge d'accueil s'ouvre en premier, choisissez Create rule (Créer une règle).
3. Saisissez un nom et une description pour la règle.

Une règle ne peut pas avoir le même nom qu'une autre règle de la même région et sur le même bus d'événement.

4. Pour Define pattern (Définir un modèle), choisissez Event pattern (Modèle d'événement).
5. Pour le modèle de correspondance d'événement, choisissez Modèle personnalisé.
6. Pour le modèle d'événement, collez le contenu suivant dans la zone :

```
{
    "source": [
        "aws.ssm"
    ],
    "detail-type": [
        "Parameter Store Change"
    ],
    "detail": {
        "name": [
            "parameter-1-name",
            "/parameter-2-name/level-2",
            "/parameter-3-name/level-2/level-3"
        ],
        "operation": [
            "Create",
            "Update",
            "Delete",
            "LabelParameterVersion"
        ]
    }
}
```

7. Modifiez le contenu pour les paramètres et les opérations sur lesquels vous voulez agir.

   Par exemple, le contenu suivant signifie qu'une action est effectuée lorsque l'un des paramètres nommés `/Oncall` et `/Project/Teamlead` est mis à jour :

```
{
    "source": [
        "aws.ssm"
    ],
    "detail-type": [
        "Parameter Store Change"
    ],
    "detail": {
        "name": [
            "/Oncall",
            "/Project/Teamlead"
        ],
        "operation": [
            "Update"
        ]
    }
}
```

8. Choisissez Save.
9. Pour Select event bus (Sélectionner un bus d'événement), choisissez le bus d'événement que vous souhaitez associer à cette règle. Si vous souhaitez que cette règle se déclenche sur les événements correspondants provenant de votre propre compte AWS, sélectionnez AWS default event bus (Bus d'événement par défaut AWS). Lorsqu'un service AWS de votre compte émet un événement, il accède toujours au bus d'événement par défaut de votre compte.
10. Pour Select targets (Sélectionner les cibles), choisissez un type de cible et une ressource prise en charge. Par exemple, si vous choisissez une rubrique SNS, sélectionnez Rubrique. Si vous choisissez CodePipeline , sélectionnez un ARN de pipeline.

11. Déployez Configure input (Configurer l'entrée) et choisissez une option. Ensuite, fournissez d'autres détails de configuration requis par le type de cible que vous avez sélectionné.
12. (Facultatif) Entrez une ou plusieurs balises pour la règle. Pour de plus amples informations, veuillez consulter Balisage de vos ressources Amazon EventBridge dans le Guide de l'utilisateur Amazon EventBridge.
13. Sélectionnez Create.

Configuration de EventBridge pour les stratégies de paramètres

Cette rubrique explique comment créer des règles EventBridge qui appellent des cibles basées sur les événements qui se produisent dans une ou plusieurs stratégies de paramètres dans votre compte AWS. Lorsque vous créez un paramètre avancé, vous spécifiez quand un paramètre expire, quand recevoir une notification avant qu'un paramètre n'expire et combien de temps attendre avant d'envoyer une notification indiquant qu'un paramètre n'a pas changé. La procédure suivante vous permet de configurer la notification pour ces événements. Pour de plus amples informations, veuillez consulter Affectation de stratégies de paramètres (p. 273) et Gestion des niveaux de paramètres (p. 228).

Pour configurer les stratégies EventBridge pour le paramètre Systems Manager

1. Ouvrez la console Amazon EventBridge à l'adresse https://console.aws.amazon.com/events/.
2. Dans le volet de navigation, choisissez Rules (Règles), puis Create rule (Créer une règle).

   -ou-

   Si la page Amazon EventBridge d'accueil s'ouvre d'abord, choisissez Create rule (Créer une règle).
3. Saisissez un nom et une description pour la règle.

   Une règle ne peut pas avoir le même nom qu'une autre règle de la même région et sur le même bus d'événement.
4. Pour Define pattern (Définir un modèle), choisissez Event pattern (Modèle d'événement).
5. Pour Modèle de correspondance d'événement, choisissez Modèle personnalisé.
6. Pour le modèle d'événement, collez le contenu suivant dans la zone :

```
{
    "source": [
        "aws.ssm"
    ],
    "detail-type": [
        "Parameter Store Policy Action"
    ],
    "detail": {
        "parameter-name": [
            "parameter-1-name",
            "/parameter-2-name/level-2",
            "/parameter-3-name/level-2/level-3"
        ],
        "policy-type": [
            "Expiration",
            "ExpirationNotification",
            "NoChangeNotification"
        ]
    }
}
```

7. Modifiez le contenu des paramètres et des types de stratégie sur lesquels vous voulez agir. Par exemple, le contenu suivant signifie qu'une action est effectuée chaque fois que le paramètre nommé `/OncallDuties` arrive à expiration et est supprimé :

```
{
    "source": [
        "aws.ssm"
    ],
    "detail-type": [
        "Parameter Store Policy Action"
    ],
    "detail": {
        "parameter-name": [
            "/OncallDuties"
        ],
        "policy-type": [
            "Expiration"
        ]
    }
}
```

8. Choisissez Save.
9. Pour Select event bus (Sélectionner un bus d'événement), choisissez le bus d'événement que vous souhaitez associer à cette règle. Si vous souhaitez que cette règle se déclenche sur les événements correspondants provenant de votre propre compte AWS, sélectionnez AWS default event bus (Bus d'événement par défaut AWS). Lorsqu'un service AWS de votre compte émet un événement, il accède toujours au bus d'événement par défaut de votre compte.
10. Pour Select targets (Sélectionner les cibles), choisissez un type de cible et une ressource prise en charge. Par exemple, si vous choisissez une rubrique SNS, sélectionnez Rubrique. Si vous choisissez CodePipeline , sélectionnez un ARN de pipeline.
11. Déployez Configure input (Configurer l'entrée) et choisissez une option. Ensuite, fournissez d'autres détails de configuration requis par le type de cible que vous avez sélectionné.
12. (Facultatif) Entrez une ou plusieurs balises pour la règle. Pour de plus amples informations, veuillez consulter Balisage de vos ressources Amazon EventBridge dans le Guide de l'utilisateur Amazon EventBridge.
13. Sélectionnez Create.

Informations connexes

- (Billet de blog) Utiliser les étiquettes de paramètre pour faciliter la mise à jour de la configuration entre les environnements
- Didacticiel : Utilisez EventBridge pour relayer des événements vers AWS Systems Manager Fonctionnalité Exécuter la commande dans leGuide de l'utilisateur Amazon EventBridge
- Didacticiel : Définir AWS Systems Manager Automation en tant que EventBridge cible dans leGuide de l'utilisateur Amazon EventBridge

Utilisation des Parameter Store

Cette section décrit comment organiser et créer des paramètres de balises, et créer différentes versions de paramètres. Vous pouvez utiliser la console AWS Systems Manager, la console Amazon EC2 ou l'interface de ligne de commande AWS pour créer et utiliser des paramètres. Pour plus d'informations sur les paramètres, consultez .Qu'est-ce qu'un paramètre ? (p. 220)

Rubriques

- Création de paramètres Systems Manager (p. 243)
- Appel de paramètres publics (p. 259)
- Affectation de stratégies de paramètres (p. 273)

- Recherche de paramètres Systems Manager (p. 279)
- Utilisation des paramètres à l'aide Fonctionnalité Exécuter la commande des commandes de l' (p. 281)
- Utilisation des hiérarchies de paramètres (p. 285)
- Utilisation des étiquettes de paramètre (p. 290)
- Utilisation des versions de paramètre (p. 296)
- Prise en charge des paramètres natifs pour Amazon Machine ImageIDs (p. 298)

Création de paramètres Systems Manager

Aidez-vous des informations des rubriques suivantes pour créer Systems Manager des paramètres à l'aide de la AWS Systems Manager console , de l' AWS Command Line Interface (AWS CLI) ou Outils AWS pour Windows PowerShell d' .

Cette section montre comment créer, stocker et exécuter des paramètres avec Parameter Store dans un environnement de test. Il montre également comment utiliser Parameter Store avec d'autres Systems Manager fonctionnalités et AWS services . Pour plus d'informations, veuillez consulter Qu'est-ce qu'un paramètre ? (p. 220)

À propos des exigences et des contraintes relatives aux noms de paramètres

Utilisez les informations indiquées dans cette rubrique pour vous aider à spécifier les valeurs valides des noms de paramètre lorsque vous créez un paramètre.

Ces informations complètent celles de la rubrique PutParameter du Référence d'API AWS Systems Manager , qui fournit également des informations sur les valeurs AllowedPattern , Description, KeyId , Overwrite, Type et Value.

Les exigences et contraintes relatives aux noms de paramètre sont les suivantes :

- Sensibilité à la casse : Les noms des paramètres sont sensibles à la casse.
- Espaces : Les noms de paramètre ne peuvent pas inclure d'espaces.
- Caractères valides : Les noms de paramètre peuvent uniquement comporter les lettres et symboles suivants :`a-zA-Z0-9_.-/`
- Format AMI valide : Lorsque vous choisissez `aws:ec2:image` comme type de données pour un `String` paramètre, l'ID que vous saisissez doit être validé pour le format d'ID d'AMI`ami-12345abcdeEXAMPLE`.
- Complètement qualifié : Lorsque vous créez ou référencez un paramètre dans une hiérarchie, vous devez inclure une barre oblique (/). Lorsque vous référencez un paramètre faisant partie d'une hiérarchie, vous devez spécifier le chemin de hiérarchie entier, y compris la barre oblique initiale (/).
  - Noms de paramètre complets : `MyParameter1`, `/MyParameter2` ,`/Dev/Production/East/Project-ABC/MyParameter`
  - Nom du paramètre non complet : `MyParameter3/L1`
- Longueur : La longueur maximale d'un nom de paramètre, y compris le contenu complet du paramètre Amazon Resource Name (ARN), est de 1 011 caractères. Par exemple, la longueur du paramètre suivant est de 65 caractères et non de 20 caractères :

```
arn:aws:ssm:us-east-2:111122223333:parameter/ExampleParameterName
```

- Préfixes : Un nom de paramètre ne peut pas être préfixé par « aws » ou « ssm » (non sensible à la casse). Par exemple, les tentatives de création de paramètres avec les noms suivants échouent avec une exception :

- `awsTestParameter`
- `SSM-testparameter`
- `/aws/testparam1`

  Note

  Lorsque vous spécifiez un paramètre dans un document, une commande ou un script SSM, incluez `ssm` dans la syntaxe, comme le montrent les exemples suivants.

  - Valide : {{ssm:*parameter-name*}} et {{ ssm :*parameter-name* }}, par exemple `{{ssm:MyParameter}}` , et`{{ ssm:MyParameter }}`

- Unicité : Un nom de paramètre doit être unique dans une AWS région . Par exemple, Systems Manager traite les paramètres suivants comme des paramètres distincts, s'ils existent dans la même région :
  - `/Test/TestParam1`
  - `/TestParam1`

  Les exemples suivants sont également uniques :
  - `/Test/TestParam1/Logpath1`
  - `/Test/TestParam1`

  Cependant, les exemples suivants, s'ils se trouvent dans la même région, ne sont pas uniques :
  - `/TestParam1`
  - `TestParam1`
- Profondeur de la hiérarchie : Si vous spécifiez une hiérarchie de paramètres, la hiérarchie peut avoir une profondeur maximale de quinze niveaux. Vous pouvez définir un paramètre à n'importe quel niveau de la hiérarchie. Les deux exemples suivants sont valides du point de vue de la structure :
  - `/Level-1/L2/L3/L4/L5/L6/L7/L8/L9/L10/L11/L12/L13/L14/parameter-name`
  - `parameter-name`

  La tentative de création du paramètre suivant échouerait avec une exception `HierarchyLevelLimitExceededException` :
  - `/Level-1/L2/L3/L4/L5/L6/L7/L8/L9/L10/L11/L12/L13/L14/L15/L16/parameter-name`

  Important

  Si un utilisateur a accès à un chemin, il peut accéder à tous les niveaux de ce chemin. Par exemple, si un utilisateur a l'autorisation d'accéder au chemin `/a` , il peut également accéder à `/a/b` . Même si un utilisateur s'est vu refuser explicitement l'accès au paramètre dans IAM , il peut toujours appeler l`/a/b`GetParametersByPath'action d'API de manière récursive pour et afficher `/a` ./a/b

Rubriques

- Créer un paramètre Systems Manager (console) (p. 244)
- Créer un paramètre Systems Manager (AWS CLI) (p. 246)
- Créer un paramètre Systems Manager (Outils pour Windows PowerShell) (p. 256)

Créer un paramètre Systems Manager (console)

Vous pouvez utiliser la AWS Systems Manager console pour créer et exécuter des types `String` de `StringList` paramètres , et .`SecureString`

Note

Les paramètres sont uniquement disponibles dans la région AWS où ils ont été créés.

Créer un paramètre (console)

La procédure suivante vous guide à travers le processus de création d'un paramètre dans la Parameter Store console . Vous pouvez créer des types `String` de `StringList` paramètres , et dans la console .`SecureString`

Pour créer un paramètre

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Parameter Store.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Parameter Store.
3. Choisissez Create parameter (Créer un paramètre).
4. Dans la zone Name (Nom), entrez une hiérarchie et un nom. Par exemple, entrez **`/Test/helloWorld`**.

   Pour plus d'informations sur les hiérarchies de paramètres, consultez Utilisation des hiérarchies de paramètres (p. 285).
5. Dans la zone Description, entrez une description qui identifie ce paramètre comme paramètre de test.
6. Pour Parameter tier (Niveau du paramètre), choisissez Standard ou Advanced (Avancé). Pour de plus amples informations sur les paramètres avancés, veuillez consulter Gestion des niveaux de paramètres (p. 228).
7. Pour Type, choisissez String, ou StringList .SecureString

   - Si vous choisissez String (Chaîne), le champ Data type (Type de données) apparaît. Si vous créez un paramètre pour contenir l'ID de ressource d'une Amazon Machine Image (AMI), sélectionnez `aws:ec2:image` . Sinon, laissez la valeur par défaut `text` sélectionnée.
   - Si vous choisissez SecureString , le champ KMS Key ID s'affiche. Si vous ne fournissez pas d'ID de clé principale client (CMK) KMS, d'ARN de CMK, de nom d'alias ou d'ARN d'alias, le système utilise `alias/aws/ssm`, qui est la clé CMK gérée par AWS pour Systems Manager. Si vous ne souhaitez pas utiliser cette clé, vous pouvez utiliser une clé CMK gérée par le client. Pour plus d'informations sur AWS les gérés par et par le client, consultez CMKs Concepts AWS Key Management Service dans le AWS Key Management Service Developer Guide . Pour de plus amples informations sur Parameter Store et sur le chiffrement KMS, veuillez consulter Comment AWS Systems Manager Parameter Store utilise-t-il AWS KMS ?.
   - Lorsque vous créez un paramètre `SecureString` dans la console à l'aide du paramètre `key-id` avec un nom d'alias de clé CMK gérée par le client ou un ARN d'alias, vous devez spécifier le préfixe `alias/` avant l'alias. Voici un exemple d'ARN :

     ```
     arn:aws:kms:us-east-2:123456789012:alias/MyAliasName
     ```

     Voici un exemple de nom d'alias :

     ```
     alias/MyAliasName
     ```
8. Dans la zone Valeur, entrez une valeur. Par exemple, saisissez **`This is my first parameter`** ou **`ami-0dbf5ea29aEXAMPLE`**.

Note

Les paramètres ne peuvent pas être référencés ou imbriqués dans les valeurs d'autres paramètres. Vous ne pouvez pas inclure `{{}}` ou `{{ssm:parameter-name}}` dans une valeur de paramètre.
Si vous avez choisi SecureString , la valeur du paramètre est masquée par défaut (« ****** ») lorsque vous l'affichez ultérieurement dans l'onglet Présentation du paramètre. Choisissez Afficher pour afficher la valeur du paramètre.

9. (Facultatif) Dans la zone Tags (Balises), appliquez une ou plusieurs paires clé/valeur de balise au paramètre.

   Les balises sont des métadonnées facultatives que vous affectez à une ressource. Les balises vous permettent de classer une ressource de différentes façons, par exemple, par objectif, par propriétaire ou par environnement. Par exemple, vous pouvez baliser un paramètre Systems Manager pour identifier le type de ressource à laquelle il s'applique, l'environnement ou le type de données de configuration référencé par le paramètre. Dans ce cas, vous pouvez spécifier les paires clé/valeur suivantes :

   - `Key=Resource,Value=S3bucket`
   - `Key=OS,Value=Windows`
   - `Key=ParameterType,Value=LicenseKey`
10. Choisissez Create parameter (Créer un paramètre).
11. Dans la liste des paramètres, sélectionnez le nom du paramètre que vous venez de créer. Vérifiez les détails dans l'onglet Overview (Présentation). Si vous avez créé un paramètre `SecureString`, choisissez Show (Afficher) pour voir la valeur non chiffrée.

    Note

    Vous ne pouvez pas remplacer un paramètre avancé par un paramètre standard. Si vous n'avez plus besoin d'un paramètre avancé ou si vous ne souhaitez plus payer de frais supplémentaires pour un paramètre avancé, vous devez le supprimer et le recréer en tant que nouveau paramètre standard.

Créer un paramètre Systems Manager (AWS CLI)

Vous pouvez utiliser l' AWS Command Line Interface (AWS CLI) pour créer des types `String` de `StringList` paramètres , et .`SecureString`

Note

Les paramètres sont uniquement disponibles dans la région AWS où ils ont été créés.
Les paramètres ne peuvent pas être référencés ou imbriqués dans les valeurs d'autres paramètres. Vous ne pouvez pas inclure `{{}}` ou `{{ssm:parameter-name}}` dans une valeur de paramètre.

Rubriques

- Créer un paramètre String (AWS CLI) (p. 247)
- Créer un paramètre StringList (AWS CLI) (p. 250)

- Créer un paramètre SecureString (AWS CLI) (p. 251)
- Créer un paramètre sur plusieurs lignes (AWS CLI) (p. 256)

Créer un paramètre `String` (AWS CLI)

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Exécutez la commande suivante pour créer un paramètre de type `String`.

   Linux

   ```
   aws ssm put-parameter \
       --name "parameter-name" \
       --value "parameter-value" \
       --type String \
       --tags "Key=tag-key,Value=tag-value"
   ```

   Windows

   ```
   aws ssm put-parameter ^
       --name "parameter-name" ^
       --value "parameter-value" ^
       --type String ^
       --tags "Key=tag-key,Value=tag-value"
   ```

   -ou-

   Exécutez la commande suivante pour créer un paramètre qui contient un ID d'AMI (Amazon Machine Image) comme valeur de paramètre.

   Linux

   ```
   aws ssm put-parameter \
       --name "parameter-name" \
       --value "an-AMI-id" \
       --type String \
       --data-type "aws:ec2:image" \
       --tags "Key=tag-key,Value=tag-value"
   ```

   Windows

   ```
   aws ssm put-parameter ^
       --name "parameter-name" ^
       --value "an-AMI-id" ^
       --type String ^
       --data-type "aws:ec2:image" ^
       --tags "Key=tag-key,Value=tag-value"
   ```

   L'option `--name` prend en charge les hiérarchies. Pour plus d'informations sur les hiérarchies, consultez Utilisation des hiérarchies de paramètres (p. 285).

   L'option `--data-type` doit être spécifiée uniquement si vous créez un paramètre contenant un ID d'AMI. Elle vérifie que la valeur du paramètre que vous entrez est un ID d'AMI Amazon Elastic

Compute Cloud (Amazon EC2) correctement formaté. Pour tous les autres paramètres, le type de données par défaut est `text` et il est facultatif de spécifier une valeur. Pour plus d'informations, consultez la section Prise en charge des paramètres natifs pour Amazon Machine ImageIDs (p. 298).

> **Important**
>
> En cas de réussite, la commande renvoie le numéro de version du paramètre. Exception : Si vous avez spécifié `aws:ec2:image` comme type de données, un nouveau numéro de version dans la réponse ne signifie pas que la valeur du paramètre a déjà été validée. Pour plus d'informations, consultez la section Prise en charge des paramètres natifs pour Amazon Machine ImageIDs (p. 298).

Dans l'exemple suivant, on ajoute deux balises de paires clé-valeur à un paramètre.

Linux

```
aws ssm put-parameter \
    --name parameter-name \
    --value "parameter-value" \
    --type "String" \
    --tags '[{"Key":"Region","Value":"East"},{"Key":"Environment",
 "Value":"Production"}]'
```

Windows

```
aws ssm put-parameter ^
    --name parameter-name ^
    --value "parameter-value" ^
    --type "String" ^
    --tags [{\"Key\":\"Region1\",\"Value\":\"East1\"},{\"Key\":\"Environment1\",
\"Value\":\"Production1\"}]
```

L'exemple suivant utilise une hiérarchie de paramètres dans le nom pour créer un `String` paramètre en texte brut. Elle renvoie le numéro de version du paramètre. Pour plus d'informations sur les hiérarchies de paramètres, consultez Utilisation des hiérarchies de paramètres (p. 285).

Linux

Paramètre ne figurant pas dans une hiérarchie

```
aws ssm put-parameter \
    --name "golden-ami" \
    --type "String" \
    --value "ami-12345abcdeEXAMPLE"
```

Paramètre dans une hiérarchie

```
aws ssm put-parameter \
    --name "\amis\linux\golden-ami" \
    --type "String" \
    --value "ami-12345abcdeEXAMPLE"
```

Windows

Paramètre ne figurant pas dans une hiérarchie

```
aws ssm put-parameter ^
```

```
    --name "golden-ami" ^
    --type "String" ^
    --value "ami-12345abcdeEXAMPLE"
```

Paramètre dans une hiérarchie

```
aws ssm put-parameter ^
    --name "\amis\windows\golden-ami" ^
    --type "String" ^
    --value "ami-12345abcdeEXAMPLE"
```

3. Exécutez la commande suivante pour afficher la dernière valeur du paramètre et vérifier les détails de votre nouveau paramètre.

```
aws ssm get-parameters --names "/Test/IAD/helloWorld"
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
    "InvalidParameters": [],
    "Parameters": [
        {
            "Name": "/Test/IAD/helloWorld",
            "Type": "String",
            "Value": "My updated parameter value",
            "Version": 2,
            "LastModifiedDate": "2020-02-25T15:55:33.677000-08:00",
            "ARN": "arn:aws:ssm:us-east-2:123456789012:parameter/Test/IAD/helloWorld"

        }
    ]
}
```

Exécutez la commande suivante pour modifier la valeur du paramètre. Elle renvoie le numéro de version du paramètre.

```
aws ssm put-parameter --name "/Test/IAD/helloWorld" --value "My updated 1st parameter" --
type String --overwrite
```

Exécutez la commande suivante pour afficher l'historique des valeurs du paramètre.

```
aws ssm get-parameter-history --name "/Test/IAD/helloWorld"
```

Exécutez la commande suivante pour utiliser ce paramètre dans une commande.

```
aws ssm send-command --document-name "AWS-RunShellScript" --parameters '{"commands":["echo
 {{ssm:/Test/IAD/helloWorld}}"]}' --targets "Key=instanceids,Values=instance-ids"
```

Exécutez la commande suivante si vous souhaitez uniquement récupérer la valeur du paramètre.

```
aws ssm get-parameter --name testDataTypeParameter --query "Parameter.Value"
```

Exécutez la commande suivante si vous souhaitez uniquement récupérer la valeur du paramètre à l'aide de `get-parameters` .

```
aws ssm get-parameters --names "testDataTypeParameter" --query "Parameters[*].Value"
```

Exécutez la commande suivante pour afficher les métadonnées du paramètre.

```
aws ssm describe-parameters --filters "Key=Name,Values=/Test/IAD/helloWorld"
```

> **Note**
>
> Nom doit être en majuscule.

Le système renvoie des informations similaires à celles qui suivent.

```
{
    "Parameters": [
        {
            "Name": "helloworld",
            "Type": "String",
            "LastModifiedUser": "arn:aws:iam::123456789012:user/user-name",
            "LastModifiedDate": 1494529763.156,
            "Version": 1,
            "Tier": "Standard",
            "Policies": []
        }
    ]
}
```

Créer un paramètre `StringList` (AWS CLI)

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Exécutez la commande suivante pour créer un paramètre.

   Linux

   ```
   aws ssm put-parameter \
       --name "parameter-name" \
       --value "a-comma-separated-list-of-values" \
       --type StringList \
       --tags "Key=tag-key,Value=tag-value"
   ```

   Windows

   ```
   aws ssm put-parameter ^
       --name "parameter-name" ^
       --value "a-comma-separated-list-of-values" ^
       --type StringList ^
       --tags "Key=tag-key,Value=tag-value"
   ```

   > **Note**
   >
   > En cas de réussite, la commande renvoie le numéro de version du paramètre.

   Cet exemple ajoute deux balises de paires clé-valeur à un paramètre. (En fonction du type de système d'exploitation sur votre ordinateur local, exécutez l'une des commandes suivantes. La version à

exécuter à partir d'une machine Windows locale inclut les caractères d'échappement (« \\ ») dont vous avez besoin pour exécuter la commande à partir de votre outil de ligne de commande.)

Voici un exemple de `StringList` qui utilise une hiérarchie de paramètres.

Linux

```
aws ssm put-parameter \
    --name /IAD/ERP/Oracle/addUsers \
    --value "Milana,Mariana,Mark,Miguel" \
    --type StringList
```

Windows

```
aws ssm put-parameter ^
    --name /IAD/ERP/Oracle/addUsers ^
    --value "Milana,Mariana,Mark,Miguel" ^
    --type StringList
```

> **Note**
>
> Les éléments d'un `StringList` doivent être séparés par une virgule (,). Vous ne pouvez pas utiliser d'autres caractères de ponctuation ou spéciaux pour introduire des articles dans la liste. Si vous avez une valeur de paramètre qui nécessite une virgule, alors utilisez le type `String`.

3. Exécutez la commande `get-parameters` pour vérifier les détails du paramètre. Exemple :

```
aws ssm get-parameters --name "/IAD/ERP/Oracle/addUsers"
```

Créer un paramètre SecureString (AWS CLI)

Utilisez la procédure suivante pour créer un paramètre `SecureString`.

> **Important**
>
> Seule la valeur d'un paramètre `SecureString` est chiffrée. Les noms de paramètre, les descriptions et les autres propriétés ne sont pas chiffrés.

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Exécutez l'une des commandes suivantes pour créer un paramètre utilisant le type de données `SecureString`.

   Linux

   Créer un **`SecureString`** paramètre à l'aide de la clé AWS gérée par défaut

```
aws ssm put-parameter \
    --name "parameter-name" \
    --value "parameter-value" \
    --type "SecureString"
```

   Créer un **`SecureString`** paramètre qui utilise une clé principale client (CMK) gérée par le client

```
aws ssm put-parameter \
    --name "parameter-name" \
    --value "a-parameter-value, for example P@ssW%rd#1" \
    --type "SecureString"
    --tags "Key=tag-key,Value=tag-value"
```

Créer un **`SecureString`** paramètre qui utilise une AWS KMS clé personnalisée

```
aws ssm put-parameter \
    --name "parameter-name" \
    --value "a-parameter-value, for example P@ssW%rd#1" \
    --type "SecureString" \
    --key-id "your-AWS-user-account-ID/the-custom-AWS KMS-key" \
    --tags "Key=tag-key,Value=tag-value"
```

Windows

Créer un **`SecureString`** paramètre à l'aide de la clé AWS gérée par défaut

```
aws ssm put-parameter ^
    --name "parameter-name" ^
    --value "parameter-value" ^
    --type "SecureString"
```

Créer un **`SecureString`** paramètre qui utilise une clé principale client (CMK) gérée par le client

```
aws ssm put-parameter ^
    --name "parameter-name" ^
    --value "a-parameter-value, for example P@ssW%rd#1" ^
    --type "SecureString" ^
    --tags "Key=tag-key,Value=tag-value"
```

Créer un **`SecureString`** paramètre qui utilise une AWS KMS clé personnalisée

```
aws ssm put-parameter ^
    --name "parameter-name" ^
    --value "a-parameter-value, for example P@ssW%rd#1" ^
    --type "SecureString" ^
    --key-id " ^
    --tags "Key=tag-key,Value=tag-value"your-AWS-user-account-ID/the-custom-AWS
 KMS-key"
```

Si vous créez un paramètre `SecureString` à l'aide de la clé AWS KMS gérée par AWS dans votre compte et votre région, il n'est pas nécessaire de fournir une valeur pour le paramètre `--key-id`.

Note

Pour utiliser la clé principale client AWS Key Management Service (clé CMK) (KMS) attribuée à votre compte et votre région, supprimez le `key-id` paramètre de la commande. Pour plus d'informations sur CMKs , consultez Concepts AWS Key Management Service dans le AWS Key Management Service Developer Guide .

Pour utiliser une clé CMK gérée par le client au lieu de la clé CMK gérée par AWS et affectée à votre compte, vous devez spécifier la clé à l'aide du paramètre `--key-id`. Le paramètre prend en charge les formats de paramètre KMS suivants.

- Exemple d'ARN de clé :

  ```
  arn:aws:kms:us-
  east-2:123456789012:key/12345678-1234-1234-1234-123456789012
  ```

- Exemple d'ARN de l'alias:

  ```
  arn:aws:kms:us-east-2:123456789012:alias/MyAliasName
  ```

- Exemple d'ID de clé :

  ```
  12345678-1234-1234-1234-123456789012
  ```

- Exemple de nom d'alias :

  ```
  alias/MyAliasName
  ```

Vous pouvez créer une clé CMK gérée par le client à l'aide de l'AWS Management Console ou de l'API AWS KMS. Les commandes de l'AWS CLI suivantes créent une clé gérée par le client dans la région actuelle de votre compte AWS.

```
aws kms create-key
```

Utilisez une commande au format suivant pour créer un paramètre `SecureString` à l'aide de la clé que vous venez de créer.

L'exemple suivant utilise un nom brouillé (`3l3vat3131`) comme paramètre de mot de passe et une clé CMK.

Linux

```
aws ssm put-parameter \
    --name /Finance/Payroll/3l3vat3131 \
    --value "P@sSwW)rd" \
    --type SecureString \
    --key-id arn:aws:kms:us-
east-2:123456789012:key/1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d5e
```

Windows

```
aws ssm put-parameter ^
    --name /Finance/Payroll/3l3vat3131 ^
    --value "P@sSwW)rd" ^
    --type SecureString ^
    --key-id arn:aws:kms:us-
east-2:123456789012:key/1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d5e
```

3. Exécutez la commande suivante pour vérifier les détails du paramètre.

   Linux

   ```
   aws ssm get-parameters \
       --name "the-parameter-name-you-specified" \
       --with-decryption
   ```

   Windows

   ```
   aws ssm get-parameters ^
       --name "the-parameter-name-you-specified" ^
   ```

```
    --with-decryption
```

Note

Si vous ne spécifiez le paramètre `with-decryption`, ou si vous spécifiez le paramètre `no-with-decryption`, la commande renvoie un GUID chiffré.

4. Exécutez la commande suivante pour afficher les métadonnées du paramètre.

   Linux

```
aws ssm describe-parameters \
    --filters "Key=Name,Values=the-name-that-you-specified"
```

   Windows

```
aws ssm describe-parameters ^
    --filters "Key=Name,Values=the-name-that-you-specified"
```

5. Exécutez la commande suivante pour modifier la valeur du paramètre si vous n'utilisez pas de clé principale client (CMK) gérée par le client.

   Linux

```
aws ssm put-parameter \
    --name "the-name-that-you-specified" \
    --value "a-new-parameter-value" \
    --type "SecureString" \
    --overwrite
```

   Windows

```
aws ssm put-parameter ^
    --name "the-name-that-you-specified" ^
    --value "a-new-parameter-value" ^
    --type "SecureString" ^
    --overwrite
```

   -ou-

   Exécutez l'une des commandes suivantes pour modifier la valeur du paramètre si vous utilisez une clé principale client (CMK) gérée par le client.

   Linux

```
aws ssm put-parameter \
    --name "the-name-that-you-specified" \
    --value "a-new-parameter-value" \
    --type "SecureString" \
    --key-id "the-CMK-ID" \
    --overwrite
```

```
aws ssm put-parameter \
    --name "the-name-that-you-specified" \
    --value "a-new-parameter-value" \
    --type "SecureString" \
```

```
    --key-id "your-AWS-user-account-alias/the-CMK-ID" \
    --overwrite
```

Windows

```
aws ssm put-parameter ^
    --name "the-name-that-you-specified" ^
    --value "a-new-parameter-value" ^
    --type "SecureString" ^
    --key-id "the-CMK-ID" ^
    --overwrite
```

```
aws ssm put-parameter ^
    --name "the-name-that-you-specified" ^
    --value "a-new-parameter-value" ^
    --type "SecureString" ^
    --key-id "your-AWS-user-account-alias/the-CMK-ID" ^
    --overwrite
```

6. Exécutez la commande suivante pour afficher la dernière valeur du paramètre.

   Linux

```
aws ssm get-parameters \
    --name "the-name-that-you-specified" \
    --with-decryption
```

   Windows

```
aws ssm get-parameters ^
    --name "the-name-that-you-specified" ^
    --with-decryption
```

7. Exécutez la commande suivante pour afficher l'historique des valeurs du paramètre.

   Linux

```
aws ssm get-parameter-history \
    --name "the-name-that-you-specified"
```

   Windows

```
aws ssm get-parameter-history ^
    --name "the-name-that-you-specified"
```

   Note

   Vous pouvez créer manuellement un paramètre avec une valeur chiffrée. Dans ce cas, dans la mesure où la valeur est déjà chiffrée, vous n'avez pas à choisir le type de paramètre `SecureString`. Si vous choisissez `SecureString`, votre paramètre sera doublement chiffré.

Par défaut, toutes les valeurs `SecureString` sont affichées sous forme de texte chiffré. Pour déchiffrer une valeur `SecureString`, un utilisateur doit avoir l'autorisation d'appeler l'action d'API Decrypt KMS. Pour plus d'informations sur la configuration du contrôle d'accès KMS, consultez Authentification et contrôle d'accès pour AWS KMS dans le AWS Key Management Service Developer Guide.

Créer un paramètre sur plusieurs lignes (AWS CLI)

Vous pouvez utiliser l' AWS CLI pour créer un paramètre avec des sauts de ligne. L'ajout de sauts de ligne vous permet de fractionner le texte en valeurs de paramètres plus longues pour une meilleure lisibilité ou, par exemple, de mettre à jour plus facilement le contenu des paramètres multiparagraphes d'une page web. Vous pouvez inclure le contenu dans un fichier JSON et utiliser l'`--cli-input-json`option, à l'aide de caractères de saut de ligne comme `/n` , comme illustré dans l'exemple suivant.

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).
2. Exécutez la commande suivante pour créer un paramètre sur plusieurs lignes.

   Linux

   ```
   aws ssm put-parameter \
       --name "MultiLineParameter" \
       --type String \
       --cli-input-json file://MultiLineParameter.json
   ```

   Windows

   ```
   aws ssm put-parameter ^
       --name "MultiLineParameter" ^
       --type String ^
       --cli-input-json file://MultiLineParameter.json
   ```

   L'exemple suivant affiche le contenu d'un fichier `MultiLineParameter.json`.

   ```
   {
       "Value": "<para>Paragraph One</para>\n<para>Paragraph Two</para>\n<para>Paragraph
    Three</para>"
   }
   ```

La valeur du paramètre enregistré est stockée comme suit.

```
<para>Paragraph One</para>
<para>Paragraph Two</para>
<para>Paragraph Three</para>
```

Créer un paramètre Systems Manager (Outils pour Windows PowerShell)

Vous pouvez utiliser Outils pour Windows PowerShell pour créer des types de paramètres `String`, `StringList` et `SecureString`.

> **Note**
>
> Les paramètres ne peuvent pas être référencés ou imbriqués dans les valeurs d'autres paramètres. Vous ne pouvez pas inclure `{{}}` ou `{{ssm:parameter-name}}` dans une valeur de paramètre.
> Les paramètres sont uniquement disponibles dans la région AWS où ils ont été créés.

Rubriques

- Créer un paramètre String (Outils pour Windows PowerShell) (p. 257)

- Créer un paramètre StringList (Outils pour Windows PowerShell) (p. 257)
- Créer un paramètre SecureString (Outils pour Windows PowerShell) (p. 258)

Créer un paramètre `String` (Outils pour Windows PowerShell)

1. Si vous ne l'avez pas déjà fait, installez et configurez l'Outils AWS pour PowerShell.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Exécutez la commande suivante pour créer un paramètre contenant une valeur de texte brut.

```
Write-SSMParameter `
    -Name "parameter-name" `
    -Value "parameter-value" `
    -Type "String"
```

   -ou-

   Exécutez la commande suivante pour créer un paramètre qui contient un ID d'AMI (Amazon Machine Image) comme valeur de paramètre.

```
Write-SSMParameter `
    -Name "parameter-name" `
    -Value "an-AMI-id" `
    -Type "String" `
    -DataType "aws:ec2:image" `
    -Tags "Key=tag-key,Value=tag-value"
```

   L'option `-DataType` doit être spécifiée uniquement si vous créez un paramètre contenant un ID d'AMI. Pour tous les autres paramètres, le type de données par défaut est `text` . Pour plus d'informations, consultez Prise en charge des paramètres natifs pour Amazon Machine ImageIDs (p. 298) .

   Voici un exemple de String qui utilise une hiérarchie de paramètres.

```
Write-SSMParameter `
    -Name "/IAD/Web/SQL/IPaddress" `
    -Value "99.99.99.999" `
    -Type "String" `
    Tags "Key=Region,Value=IAD"
```

3. Exécutez la commande suivante pour vérifier les détails du paramètre.

```
(Get-SSMParameterValue -Name "the-parameter-name-you-specified").Parameters
```

Créer un paramètre `StringList` (Outils pour Windows PowerShell)

1. Si vous ne l'avez pas déjà fait, installez et configurez l'Outils AWS pour PowerShell.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Exécutez la commande suivante pour créer un StringList paramètre.

```
Write-SSMParameter `
    -Name "parameter-name" `
    -Value "a-comma-separated-list-of-values" `
```

```
    -Type "StringList" `
    -Tags "Key=tag-key,Value=tag-value"
```

En cas de réussite, la commande renvoie le numéro de version du paramètre.

> **Note**
>
> Les éléments d'un `StringList` doivent être séparés par une virgule (,). Vous ne pouvez pas utiliser d'autres caractères de ponctuation ou spéciaux pour introduire des articles dans la liste. Si vous avez une valeur de paramètre qui nécessite une virgule, alors utilisez le type `String`.

3. Exécutez la commande suivante pour vérifier les détails du paramètre.

```
(Get-SSMParameterValue -Name "the-parameter-name-you-specified").Parameters
```

Créer un paramètre SecureString (Outils pour Windows PowerShell)

Avant de créer un paramètre `SecureString`, lisez les exigences liées à ce type de paramètre. Pour plus d'informations, consultez la section Créer un paramètre SecureString (AWS CLI) (p. 251).

1. Si vous ne l'avez pas déjà fait, installez et configurez l'Outils AWS pour PowerShell.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Exécutez la commande suivante pour créer un paramètre.

```
Write-SSMParameter `
    -Name "parameter-name" `
    -Value "parameter-value" `
    -Type "SecureString"  `
    -KeyId "a KMS CMK ID, a KMS CMK ARN, an alias name, or an alias ARN" `
    --tags "Key=tag-key,Value=tag-value"
```

En cas de réussite, la commande renvoie le numéro de version du paramètre.

> **Note**
>
> Pour utiliser la clé principale client (CMK) gérée par AWS affectée à votre compte, supprimez le paramètre `-KeyId` de la commande.

Voici un exemple qui utilise un nom brouillé (3l3vat3131) pour un paramètre de mot de passe et une clé principale client (CMK) gérée par AWS.

```
Write-SSMParameter `
    -Name "/Finance/Payroll/3l3vat3131" `
    -Value "P@sSwW)rd" `
    -Type "SecureString"
```

3. Exécutez la commande suivante pour vérifier les détails du paramètre.

```
(Get-SSMParameterValue -Name "the-parameter-name-you-specified" -WithDecryption
 $true).Parameters
```

Appel de paramètres publics

Certains services AWS publient des informations sur les artefacts courants en tant que paramètres publics Systems Manager. Par exemple, le service Amazon Elastic Compute Cloud (Amazon EC2) publie des informations sur les AMI (Amazon Machines Image) en tant que paramètres publics.

Vous pouvez appeler ces informations à partir de vos scripts et de votre code à l'aide des actions GetParametersByPath d'API GetParameter , et .GetParameters

Articles de blog connexes

- Requête de AWS régions, de points de terminaison, etc. avec AWS Systems Manager Parameter Store
- Interrogation de la dernière Amazon Linux AMI IDs à l'aide de AWS Systems Manager Parameter Store
- Requête portant sur la dernière AMI Windows à l'aide de AWS Systems Manager Parameter Store

Rubriques

- Appel de paramètres publics d'AMI (p. 259)
- Appel d'un paramètre public d'AMI optimisée pour ECS (p. 264)
- Appel d'un paramètre public d'AMI optimisée pour EKS (p. 264)
- Appel de paramètres publics pour les services AWS, les régions, les points de terminaison, les zones de disponibilité et les zones locales (p. 265)

Appel de paramètres publics d'AMI

Les paramètres publics Amazon EC2 de l'AMI sont disponibles pour Amazon Linux / Amazon Linux 2 et Windows Server à partir des chemins suivants :

- Amazon Linux/Amazon Linux 2: `/aws/service/ami-amazon-linux-latest`
- Windows Server: `/aws/service/ami-windows-latest`

Appel de paramètres publics d'AMI pour Amazon Linux etAmazon Linux 2

Vous pouvez afficher une liste de tous les Amazon Linux et Amazon Linux 2 AMIs de la AWS région actuelle à l'aide de la commande suivante dans l' AWS CLI .

Linux

```
aws ssm get-parameters-by-path \
    --path /aws/service/ami-amazon-linux-latest \
    --query 'Parameters[].Name'
```

Windows

```
aws ssm get-parameters-by-path ^
    --path /aws/service/ami-amazon-linux-latest ^
    --query Parameters[].Name
```

La commande renvoie des informations telles que les suivantes.

```
[
```

```
    "/aws/service/ami-amazon-linux-latest/amzn-ami-hvm-x86_64-ebs",
    "/aws/service/ami-amazon-linux-latest/amzn-ami-hvm-x86_64-gp2",
    "/aws/service/ami-amazon-linux-latest/amzn-ami-hvm-x86_64-s3",
    "/aws/service/ami-amazon-linux-latest/amzn-ami-minimal-hvm-x86_64-s3",
    "/aws/service/ami-amazon-linux-latest/amzn-ami-minimal-pv-x86_64-s3",
    "/aws/service/ami-amazon-linux-latest/amzn-ami-pv-x86_64-s3",
    "/aws/service/ami-amazon-linux-latest/amzn2-ami-hvm-arm64-gp2",
    "/aws/service/ami-amazon-linux-latest/amzn2-ami-hvm-x86_64-ebs",
    "/aws/service/ami-amazon-linux-latest/amzn2-ami-hvm-x86_64-gp2",
    "/aws/service/ami-amazon-linux-latest/amzn2-ami-minimal-hvm-arm64-ebs",
    "/aws/service/ami-amazon-linux-latest/amzn-ami-minimal-hvm-x86_64-ebs",
    "/aws/service/ami-amazon-linux-latest/amzn-ami-minimal-pv-x86_64-ebs",
    "/aws/service/ami-amazon-linux-latest/amzn-ami-pv-x86_64-ebs",
    "/aws/service/ami-amazon-linux-latest/amzn2-ami-minimal-hvm-x86_64-ebs"
]
```

Vous pouvez afficher les détails relatifs à ces AMIs , y compris l'AMI IDs et les Amazon Resource Names (ARN), à l'aide de la commande suivante.

Linux

```
aws ssm get-parameters-by-path \
    --path "/aws/service/ami-amazon-linux-latest" \
    --region region
```

Windows

```
aws ssm get-parameters-by-path ^
    --path "/aws/service/ami-amazon-linux-latest" ^
    --region region
```

région représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de *region* prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

La commande renvoie des informations telles que les suivantes. Cet exemple de sortie a été tronqué faute d'espace.

```
{
    "Parameters": [
        {
            "Name": "/aws/service/ami-amazon-linux-latest/amzn-ami-hvm-x86_64-ebs",
            "Type": "String",
            "Value": "ami-0d75cc1d706735521",
            "Version": 7,
            "LastModifiedDate": 1543873943.358,
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/ami-amazon-linux-latest/
amzn-ami-hvm-x86_64-ebs"
        },
        {
            "Name": "/aws/service/ami-amazon-linux-latest/amzn-ami-hvm-x86_64-gp2",
            "Type": "String",
            "Value": "ami-0cd3dfa4e37921605",
            "Version": 7,
            "LastModifiedDate": 1543873943.47,
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/ami-amazon-linux-latest/
amzn-ami-hvm-x86_64-gp2"
        },
        {
```

```
            "Name": "/aws/service/ami-amazon-linux-latest/amzn-ami-hvm-x86_64-s3",
            "Type": "String",
            "Value": "ami-0a0e3ff8af8d19497",
            "Version": 7,
            "LastModifiedDate": 1543873943.576,
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/ami-amazon-linux-latest/
amzn-ami-hvm-x86_64-s3"
        },
        {
            "Name": "/aws/service/ami-amazon-linux-latest/amzn-ami-minimal-hvm-x86_64-ebs",
            "Type": "String",
            "Value": "ami-0786a9626196d6dac",
            "Version": 7,
            "LastModifiedDate": 1543873943.682,
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/ami-amazon-linux-latest/
amzn-ami-minimal-hvm-x86_64-ebs"
        },
```

Vous pouvez afficher les détails d'une AMI spécifique à l'aide de l'action GetParameters d'API avec le nom d'AMI complet, y compris le chemin. Voici un exemple de commande.

Linux

```
aws ssm get-parameters \
    --names /aws/service/ami-amazon-linux-latest/amzn2-ami-hvm-x86_64-gp2 \
    --region us-west-2
```

Windows

```
aws ssm get-parameters ^
    --names /aws/service/ami-amazon-linux-latest/amzn2-ami-hvm-x86_64-gp2 ^
    --region us-west-2
```

La commande renvoie les informations suivantes.

```
{
    "Parameters": [
        {
            "Name": "/aws/service/ami-amazon-linux-latest/amzn2-ami-hvm-x86_64-gp2",
            "Type": "String",
            "Value": "ami-061392db613a6357b",
            "Version": 16,
            "LastModifiedDate": 1552519670.776,
            "ARN": "arn:aws:ssm:us-west-2::parameter/aws/service/ami-amazon-linux-latest/
amzn2-ami-hvm-x86_64-gp2"
        }
    ],
    "InvalidParameters": []
}
```

Appel de paramètres publics d'AMI pourWindows Server

Vous pouvez afficher une liste de tous les Windows Server AMIs de la AWS région actuelle à l'aide de la commande suivante dans l' AWS CLI .

Linux

```
aws ssm get-parameters-by-path \
    --path /aws/service/ami-windows-latest \
```

```
    --query 'Parameters[].Name'
```

Windows

```
aws ssm get-parameters-by-path ^
    --path /aws/service/ami-windows-latest ^
    --query Parameters[].Name
```

La commande renvoie des informations telles que les suivantes. Cet exemple de sortie a été tronqué faute d'espace.

```
[
    "/aws/service/ami-windows-latest/Windows_Server-2012-R2_RTM-Chinese_Simplified-64Bit-
Base",
    "/aws/service/ami-windows-latest/Windows_Server-2012-R2_RTM-Chinese_Traditional-64Bit-
Base",
    "/aws/service/ami-windows-latest/Windows_Server-2012-R2_RTM-Dutch-64Bit-Base",
    "/aws/service/ami-windows-latest/Windows_Server-2012-R2_RTM-Hungarian-64Bit-Base",
    "/aws/service/ami-windows-latest/Windows_Server-2012-R2_RTM-Japanese-64Bit-Base",
    "/aws/service/ami-windows-latest/Windows_Server-2016-English-Core-Containers",
    "/aws/service/ami-windows-latest/Windows_Server-2016-German-Full-Base",
    "/aws/service/ami-windows-latest/Windows_Server-2016-Japanese-Full-SQL_2017_Web",
    "/aws/service/ami-windows-latest/amzn2-ami-hvm-2.0.20190313-x86_64-gp2-
SQL_2017_Express",
    "/aws/service/ami-windows-latest/amzn2-ami-hvm-2.0.20191217.0-x86_64-gp2-mono",
    "/aws/service/ami-windows-latest/EC2LaunchV2_Preview-Windows_Server-2019-English-Core-
Base",
    "/aws/service/ami-windows-latest/Windows_Server-2012-R2_RTM-English-64Bit-
SQL_2016_SP2_Standard",
    "/aws/service/ami-windows-latest/Windows_Server-2012-RTM-Italian-64Bit-Base",
    "/aws/service/ami-windows-latest/Windows_Server-2016-English-Deep-Learning",
    "/aws/service/ami-windows-latest/Windows_Server-2016-English-Full-
SQL_2014_SP3_Enterprise",
    "/aws/service/ami-windows-latest/Windows_Server-2016-Korean-Full-
SQL_2016_SP2_Standard",
    "/aws/service/ami-windows-latest/Windows_Server-2019-English-Core-EKS_Optimized-1.17",
]
```

Vous pouvez afficher les détails relatifs à ces AMIs , y compris l'AMI IDs et les Amazon Resource Names (ARN), à l'aide de la commande suivante.

Linux

```
aws ssm get-parameters-by-path \
    --path "/aws/service//aws/service/ami-windows-latest" \
    --region region
```

Windows

```
aws ssm get-parameters-by-path ^
    --path "/aws/service//aws/service/ami-windows-latest" ^
    --region region
```

région représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de *region* prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

La commande renvoie des informations telles que les suivantes. Cet exemple de sortie a été tronqué faute d'espace.

```
{
            "Name": "/aws/service/ami-windows-latest/Windows_Server-2016-English-Core-
Containers",
            "Type": "String",
            "Value": "ami-0b19f0a8c08620b34",
            "Version": 53,
            "LastModifiedDate": "2020-09-10T19:49:05.760000-07:00",
            "ARN": "arn:aws:ssm:us-east-1::parameter/aws/service/ami-windows-latest/
Windows_Server-2016-English-Core-Containers",
            "DataType": "text"
        },
        {
            "Name": "/aws/service/ami-windows-latest/Windows_Server-2016-German-Full-Base",
            "Type": "String",
            "Value": "ami-068badb2c4fe9d831",
            "Version": 53,
            "LastModifiedDate": "2020-09-10T19:53:44.049000-07:00",
            "ARN": "arn:aws:ssm:us-east-1::parameter/aws/service/ami-windows-latest/
Windows_Server-2016-German-Full-Base",
            "DataType": "text"
        },
        {
            "Name": "/aws/service/ami-windows-latest/Windows_Server-2016-Japanese-Full-
SQL_2017_Web",
            "Type": "String",
            "Value": "ami-0b12e609d6733dac7",
            "Version": 20,
            "LastModifiedDate": "2020-09-10T19:55:24.166000-07:00",
            "ARN": "arn:aws:ssm:us-east-1::parameter/aws/service/ami-windows-latest/
Windows_Server-2016-Japanese-Full-SQL_2017_Web",
            "DataType": "text"
        },
```

Vous pouvez afficher les détails d'une AMI spécifique à l'aide de l'action GetParameters d'API avec le nom d'AMI complet, y compris le chemin. Voici un exemple de commande.

Linux

```
aws ssm get-parameters \
    --names /aws/service/ami-windows-latest/Windows_Server-2016-English-Core-Containers
 \
    --region us-west-2
```

Windows

```
aws ssm get-parameters ^
    --names /aws/service/ami-windows-latest/Windows_Server-2016-English-Core-Containers
 ^
    --region us-west-2
```

La commande renvoie les informations suivantes.

```
{
    "Parameters": [
        {
            "Name": "/aws/service/ami-windows-latest/Windows_Server-2016-English-Core-
Containers",
```

```
            "Type": "String",
            "Value": "ami-0315ac376f3dac169",
            "Version": 53,
            "LastModifiedDate": "2020-09-10T19:50:01.955000-07:00",
            "ARN": "arn:aws:ssm:us-west-2::parameter/aws/service/ami-windows-latest/
Windows_Server-2016-English-Core-Containers",
            "DataType": "text"
        }
    ],
    "InvalidParameters": []
}
```

Appel d'un paramètre public d'AMI optimisée pour ECS

Le service Amazon Elastic Container Service (Amazon ECS) publie le nom de la dernière AMI optimisée pour Amazon ECS en tant que paramètre public. Les utilisateurs sont invités à utiliser cette AMI lors de la création d'un nouveau cluster Amazon EC2 pour Amazon ECS, car l'AMI optimisée inclut des correctifs de bogues et des mises à jour de fonctions. Utilisez la commande suivante pour afficher le nom de la dernière AMI optimisée pour Amazon ECS.

Linux

```
aws ssm get-parameters \
    --names /aws/service/ecs/optimized-ami/amazon-linux/recommended
```

Windows

```
aws ssm get-parameters ^
    --names /aws/service/ecs/optimized-ami/amazon-linux/recommended
```

La commande renvoie des informations telles que les suivantes.

```
{
    "Parameters": [
        {
            "Name": "/aws/service/ecs/optimized-ami/amazon-linux/recommended",
            "Type": "String",
            "Value": "{\"schema_version\":1,\"image_name\":\"amzn-ami-2018.03.p-
amazon-ecs-optimized\",\"image_id\":\"ami-01a82c3fce2c3ba58\",\"os\":\"Amazon Linux\",
\"ecs_runtime_version\":\"Docker version 18.06.1-ce\",\"ecs_agent_version\":\"1.27.0\"}",
            "Version": 22,
            "LastModifiedDate": 1555434745.425,
            "ARN": "arn:aws:ssm:us-west-2::parameter/aws/service/ecs/optimized-ami/amazon-
linux/recommended"
        }
    ],
    "InvalidParameters": []
}
```

Appel d'un paramètre public d'AMI optimisée pour EKS

Le service Amazon Elastic Kubernetes Service (Amazon EKS) publie le nom de la dernière AMI optimisée pour Amazon EKS en tant que paramètre public. Les utilisateurs sont encouragés à utiliser cette AMI lors de l'ajout de nœuds à un cluster Amazon EKS, car les nouvelles versions incluent des correctifs Kubernetes et des mises à jour de sécurité. Auparavant, vous assurer que vous utilisiez la dernière AMI impliquait de vérifier la documentation Amazon EKS et de mettre à jour manuellement les modèles de déploiement ou les ressources avec le nouvel ID d'AMI.

Utilisez la commande suivante pour afficher le nom de la dernière AMI optimisée pour Amazon EKS.

Linux

```
aws ssm get-parameters \
    --names /aws/service/eks/optimized-ami/1.14/amazon-linux-2/recommended
```

Windows

```
aws ssm get-parameters ^
    --names /aws/service/eks/optimized-ami/1.14/amazon-linux-2/recommended
```

La commande renvoie des informations telles que les suivantes.

```
{
    "Parameters": [
        {
            "Name": "/aws/service/eks/optimized-ami/1.14/amazon-linux-2/recommended",
            "Type": "String",
            "Value": "{\"schema_version\":\"2\",\"image_id\":\"ami-0b89776dcfa5f2dee
\",\"image_name\":\"amazon-eks-node-1.14-v20200507\",\"release_version\":
\"1.14.9-20200507\"}",
            "Version": 11,
            "LastModifiedDate": "2020-05-11T17:34:51.990000-07:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/eks/optimized-ami/1.14/
amazon-linux-2/recommended"
        }
    ],
    "InvalidParameters": []
}
```

Appel de paramètres publics pour les services AWS, les régions, les points de terminaison, les zones de disponibilité et les zones locales

Vous pouvez appeler les paramètres publics du service, de la région, du point de terminaison et de la zone de disponibilité AWS à l'aide du chemin suivant.

/aws/service/global-infrastructure

Afficher les régions AWS actives

Vous pouvez afficher une liste de toutes les régions AWS actives à l'aide de la commande suivante dans l'AWS CLI.

Linux

```
aws ssm get-parameters-by-path \
    --path /aws/service/global-infrastructure/regions \
    --query 'Parameters[].Name'
```

Windows

```
aws ssm get-parameters-by-path ^
    --path /aws/service/global-infrastructure/regions ^
    --query Parameters[].Name
```

La commande renvoie des informations telles que les suivantes.

```
[
    "/aws/service/global-infrastructure/regions/af-south-1",
    "/aws/service/global-infrastructure/regions/ap-east-1",
    "/aws/service/global-infrastructure/regions/ap-northeast-3",
    "/aws/service/global-infrastructure/regions/ap-southeast-1",
    "/aws/service/global-infrastructure/regions/ca-central-1",
    "/aws/service/global-infrastructure/regions/cn-north-1",
    "/aws/service/global-infrastructure/regions/eu-west-2",
    "/aws/service/global-infrastructure/regions/eu-west-3",
    "/aws/service/global-infrastructure/regions/us-east-1",
    "/aws/service/global-infrastructure/regions/us-gov-west-1",
    "/aws/service/global-infrastructure/regions/ap-northeast-2",
    "/aws/service/global-infrastructure/regions/ap-south-1",
    "/aws/service/global-infrastructure/regions/ap-southeast-2",
    "/aws/service/global-infrastructure/regions/cn-northwest-1",
    "/aws/service/global-infrastructure/regions/eu-south-1",
    "/aws/service/global-infrastructure/regions/me-south-1",
    "/aws/service/global-infrastructure/regions/sa-east-1",
    "/aws/service/global-infrastructure/regions/us-east-2",
    "/aws/service/global-infrastructure/regions/us-gov-east-1",
    "/aws/service/global-infrastructure/regions/us-west-1",
    "/aws/service/global-infrastructure/regions/ap-northeast-1",
    "/aws/service/global-infrastructure/regions/eu-central-1",
    "/aws/service/global-infrastructure/regions/eu-north-1",
    "/aws/service/global-infrastructure/regions/eu-west-1",
    "/aws/service/global-infrastructure/regions/us-west-2"
]
```

Afficher les services AWS disponibles

Vous pouvez afficher une liste complète de tous les services AWS disponibles et les trier par ordre alphabétique à l'aide de la commande suivante. Cet exemple de sortie a été tronqué faute d'espace.

Linux

```
aws ssm get-parameters-by-path \
    --path /aws/service/global-infrastructure/services \
    --query 'Parameters[].Name | sort(@)'
```

Windows

```
aws ssm get-parameters-by-path ^
    --path /aws/service/global-infrastructure/services ^
    --query "Parameters[].Name | sort(@)"
```

La commande renvoie des informations telles que les suivantes.

```
[
    "/aws/service/global-infrastructure/services/acm-pca",
    "/aws/service/global-infrastructure/services/acm",
    "/aws/service/global-infrastructure/services/alexaforbusiness",
    "/aws/service/global-infrastructure/services/apigateway",
    "/aws/service/global-infrastructure/services/application-autoscaling",
    "/aws/service/global-infrastructure/services/appmesh",
    "/aws/service/global-infrastructure/services/appstream",
    "/aws/service/global-infrastructure/services/appsync",
    "/aws/service/global-infrastructure/services/athena",
    "/aws/service/global-infrastructure/services/autoscaling-plans",
    "/aws/service/global-infrastructure/services/autoscaling",
    "/aws/service/global-infrastructure/services/backup",
```

```
    "/aws/service/global-infrastructure/services/batch",
    "/aws/service/global-infrastructure/services/budgets",
    "/aws/service/global-infrastructure/services/ce",
    "/aws/service/global-infrastructure/services/chime",
    "/aws/service/global-infrastructure/services/cloud9",
    "/aws/service/global-infrastructure/services/cloudformation",
    "/aws/service/global-infrastructure/services/cloudfront",
    "/aws/service/global-infrastructure/services/cloudhsm",
    "/aws/service/global-infrastructure/services/cloudhsmv2",
    "/aws/service/global-infrastructure/services/cloudsearch",
    "/aws/service/global-infrastructure/services/cloudtrail",
    "/aws/service/global-infrastructure/services/cloudwatch",
    "/aws/service/global-infrastructure/services/codebuild",
```

Afficher les régions prises en charge pour un service AWS

Vous pouvez afficher la liste des régions AWS où un service est disponible. Cet exemple utilise Systems Manager (ssm).

Linux

```
aws ssm get-parameters-by-path \
    --path /aws/service/global-infrastructure/services/ssm/regions \
    --query 'Parameters[].Value'
```

Windows

```
aws ssm get-parameters-by-path ^
    --path /aws/service/global-infrastructure/services/ssm/regions ^
    --query Parameters[].Value
```

La commande renvoie des informations telles que les suivantes.

```
[
    "ap-south-1",
    "ca-central-1",
    "cn-north-1",
    "eu-central-1",
    "eu-west-1",
    "eu-west-2",
    "eu-west-3",
    "me-south-1",
    "us-east-2",
    "us-gov-west-1",
    "af-south-1",
    "ap-east-1",
    "ap-northeast-1",
    "ap-northeast-2",
    "ap-southeast-1",
    "ap-southeast-2",
    "eu-north-1",
    "eu-south-1",
    "us-gov-east-1",
    "us-west-1",
    "cn-northwest-1",
    "sa-east-1",
    "us-east-1",
    "us-west-2"
]
```

Afficher le point de terminaison régional d'un service

Vous pouvez afficher un point de terminaison régional pour un service à l'aide de la commande suivante.

Linux

```
aws ssm get-parameter \
    --name /aws/service/global-infrastructure/regions/us-west-1/services/ssm/endpoint \
    --query 'Parameter.Value'
```

Windows

```
aws ssm get-parameter ^
    --name /aws/service/global-infrastructure/regions/us-west-1/services/ssm/endpoint ^
    --query Parameter.Value
```

La commande renvoie des informations telles que les suivantes.

```
"ssm.us-west-1.amazonaws.com"
```

Afficher les détails complets de la zone de disponibilité

Vous pouvez afficher les zones de disponibilité à l'aide de la commande suivante.

Linux

```
aws ssm get-parameters-by-path \
    --path /aws/service/global-infrastructure/availability-zones/
```

Windows

```
aws ssm get-parameters-by-path ^
    --path /aws/service/global-infrastructure/availability-zones/
```

La commande renvoie des informations telles que les suivantes. Cet exemple a été tronqué faute d'espace.

```
{
    "Parameters": [
        {
            "Name": "/aws/service/global-infrastructure/availability-zones/apne3-az3",
            "Type": "String",
            "Value": "apne3-az3",
            "Version": 1,
            "LastModifiedDate": "2020-04-03T14:29:28.995000-07:00",
            "ARN": "arn:aws:ssm:us-west-2::parameter/aws/service/global-infrastructure/
availability-zones/apne3-az3"
        },
        {
            "Name": "/aws/service/global-infrastructure/availability-zones/aps1-az2",
            "Type": "String",
            "Value": "aps1-az2",
            "Version": 1,
            "LastModifiedDate": "2020-04-03T14:21:02.690000-07:00",
            "ARN": "arn:aws:ssm:us-west-2::parameter/aws/service/global-infrastructure/
availability-zones/aps1-az2"
        },
        {
            "Name": "/aws/service/global-infrastructure/availability-zones/cnn1-az2",
            "Type": "String",
```

```
            "Value": "cnn1-az2",
            "Version": 1,
            "LastModifiedDate": "2020-04-03T14:19:57.254000-07:00",
            "ARN": "arn:aws:ssm:us-west-2::parameter/aws/service/global-infrastructure/
availability-zones/cnn1-az2"
        },
```

Afficher les noms des zones de disponibilité uniquement

Vous pouvez afficher les noms des zones de disponibilité uniquement à l'aide de la commande suivante.

Linux

```
aws ssm get-parameters-by-path \
    --path /aws/service/global-infrastructure/availability-zones \
    --query 'Parameters[].Name | sort(@)'
```

Windows

```
aws ssm get-parameters-by-path ^
    --path /aws/service/global-infrastructure/availability-zones ^
    --query "Parameters[].Name | sort(@)"
```

La commande renvoie des informations telles que les suivantes. Cet exemple a été tronqué faute d'espace.

```
[
    "/aws/service/global-infrastructure/availability-zones/afs1-az1",
    "/aws/service/global-infrastructure/availability-zones/afs1-az2",
    "/aws/service/global-infrastructure/availability-zones/afs1-az3",
    "/aws/service/global-infrastructure/availability-zones/ape1-az1",
    "/aws/service/global-infrastructure/availability-zones/ape1-az2",
    "/aws/service/global-infrastructure/availability-zones/ape1-az3",
    "/aws/service/global-infrastructure/availability-zones/apne1-az1",
    "/aws/service/global-infrastructure/availability-zones/apne1-az2",
    "/aws/service/global-infrastructure/availability-zones/apne1-az3",
    "/aws/service/global-infrastructure/availability-zones/apne1-az4",
```

Afficher les noms des zones de disponibilité dans une seule région

Vous pouvez afficher les noms des zones de disponibilité dans une région (`us-east-1`, dans cet exemple) à l'aide de la commande suivante.

Linux

```
aws ssm get-parameters-by-path \
    --path /aws/service/global-infrastructure/regions/us-east-1/availability-zones \
    --query 'Parameters[].Name | sort(@)'
```

Windows

```
aws ssm get-parameters-by-path ^
    --path /aws/service/global-infrastructure/regions/us-east-1/availability-zones ^
    --query "Parameters[].Name | sort(@)"
```

La commande renvoie des informations telles que les suivantes. Cet exemple a été tronqué faute d'espace.

```
[
```

```
    "/aws/service/global-infrastructure/regions/us-east-1/availability-zones/use1-az1",
    "/aws/service/global-infrastructure/regions/us-east-1/availability-zones/use1-az2",
    "/aws/service/global-infrastructure/regions/us-east-1/availability-zones/use1-az3",
    "/aws/service/global-infrastructure/regions/us-east-1/availability-zones/use1-az4",
    "/aws/service/global-infrastructure/regions/us-east-1/availability-zones/use1-az5",
    "/aws/service/global-infrastructure/regions/us-east-1/availability-zones/use1-az6"
```

Afficher la zone de disponibilité ARNs uniquement

Vous pouvez afficher l' ARNs de zones de disponibilité uniquement à l'aide de la commande suivante.

Linux

```
aws ssm get-parameters-by-path \
    --path /aws/service/global-infrastructure/availability-zones \
    --query 'Parameters[].ARN | sort(@)'
```

Windows

```
aws ssm get-parameters-by-path ^
    --path /aws/service/global-infrastructure/availability-zones ^
    --query "Parameters[].ARN | sort(@)"
```

La commande renvoie des informations telles que les suivantes.

```
[
    "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/availability-zones/
afs1-az1",
    "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/availability-zones/
afs1-az2",
    "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/availability-zones/
afs1-az3",
    "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/availability-zones/
ape1-az1",
    "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/availability-zones/
ape1-az2",
    "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/availability-zones/
ape1-az3",
    "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/availability-zones/
apne1-az1",
```

Afficher les détails de la zone locale

Vous pouvez afficher les zones locales à l'aide de la commande suivante.

> **Note**
>
> Une seule zone locale est actuellement disponible.

Linux

```
aws ssm get-parameters-by-path \
    --path /aws/service/global-infrastructure/local-zones/
```

Windows

```
aws ssm get-parameters-by-path ^
    --path /aws/service/global-infrastructure/local-zones/
```

La commande renvoie des informations telles que les suivantes.

```
{
    "Parameters": [
        {
            "Name": "/aws/service/global-infrastructure/local-zones/usw2-lax1-az1",
            "Type": "String",
            "Value": "usw2-lax1-az1",
            "Version": 1,
            "LastModifiedDate": "2020-04-29T20:27:03.291000-07:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/
local-zones/usw2-lax1-az1"
        }
    ]
}
```

Afficher tous les paramètres et valeurs sous une zone locale

Vous pouvez afficher toutes les données de paramètres d'une zone locale à l'aide de la commande suivante.

Linux

```
aws ssm get-parameters-by-path \
    --path "/aws/service/global-infrastructure/local-zones/usw2-lax1-az1/"
```

Windows

```
aws ssm get-parameters-by-path ^
    --path "/aws/service/global-infrastructure/local-zones/usw2-lax1-az1/"
```

La commande renvoie des informations telles que les suivantes.

```
{
    "Parameters": [
        {
            "Name": "/aws/service/global-infrastructure/local-zones/usw2-lax1-az1/
geolocationCountry",
            "Type": "String",
            "Value": "US",
            "Version": 1,
            "LastModifiedDate": "2020-04-29T20:27:03.430000-07:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/
local-zones/usw2-lax1-az1/geolocationCountry"
        },
        {
            "Name": "/aws/service/global-infrastructure/local-zones/usw2-lax1-az1/
geolocationRegion",
            "Type": "String",
            "Value": "US-CA",
            "Version": 1,
            "LastModifiedDate": "2020-04-29T20:27:03.489000-07:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/
local-zones/usw2-lax1-az1/geolocationRegion"
        },
        {
            "Name": "/aws/service/global-infrastructure/local-zones/usw2-lax1-az1/network-
border-group",
            "Type": "String",
            "Value": "us-west-2-lax-1",
```

```
            "Version": 1,
            "LastModifiedDate": "2020-04-29T20:27:03.611000-07:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/
local-zones/usw2-lax1-az1/network-border-group"
        },
        {
            "Name": "/aws/service/global-infrastructure/local-zones/usw2-lax1-az1/parent-
availability-zone",
            "Type": "String",
            "Value": "usw2-az2",
            "Version": 1,
            "LastModifiedDate": "2020-04-29T20:27:03.736000-07:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/
local-zones/usw2-lax1-az1/parent-availability-zone"
        },
        {
            "Name": "/aws/service/global-infrastructure/local-zones/usw2-lax1-az1/parent-
region",
            "Type": "String",
            "Value": "us-west-2",
            "Version": 1,
            "LastModifiedDate": "2020-04-29T20:27:03.670000-07:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/
local-zones/usw2-lax1-az1/parent-region"
        },
        {
            "Name": "/aws/service/global-infrastructure/local-zones/usw2-lax1-az1/zone-
group",
            "Type": "String",
            "Value": "us-west-2-lax-1",
            "Version": 1,
            "LastModifiedDate": "2020-04-29T20:27:03.552000-07:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/
local-zones/usw2-lax1-az1/zone-group"
        }
    ]
}
```

Afficher les noms des paramètres de zone locale uniquement

Vous pouvez afficher uniquement les noms des paramètres de zone locale à l'aide de la commande suivante.

Linux

```
aws ssm get-parameters-by-path \
    --path /aws/service/global-infrastructure/local-zones/usw2-lax1-az1 \
    --query 'Parameters[].Name | sort(@)'
```

Windows

```
aws ssm get-parameters-by-path ^
    --path /aws/service/global-infrastructure/local-zones/usw2-lax1-az1 ^
    --query "Parameters[].Name | sort(@)"
```

La commande renvoie des informations telles que les suivantes.

```
[
    "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/local-zones/usw2-
lax1-az1/geolocationCountry",
```

```
    "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/local-zones/usw2-
lax1-az1/geolocationRegion",
    "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/local-zones/usw2-
lax1-az1/network-border-group",
    "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/local-zones/usw2-
lax1-az1/parent-availability-zone",
    "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/local-zones/usw2-
lax1-az1/parent-region",
    "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/local-zones/usw2-
lax1-az1/zone-group"
]
```

Affectation de stratégies de paramètres

Les stratégies de paramètre vous aident à gérer un ensemble croissant de paramètres en vous permettant de leur attribuer des critères spécifiques, tels que la date d'expiration ou la durée de vie. Les stratégies de paramètre sont particulièrement utiles pour vous obliger à mettre à jour ou supprimer les mots de passe et les données de configuration stockés dans Parameter Store . Parameter Store propose les types de stratégies suivants : `Expiration`, et `ExpirationNotification` .`NoChangeNotification` Les stratégies sont décrites plus en détail dans cette section.

> **Note**
>
> Pour implémenter des cycles de vie de renouvellement de mot de passe, utilisez AWS Secrets Manager. Secrets Manager vous permet de renouveler, gérer et récupérer facilement des informations d'identification de base de données, des clés d'API et d'autres secrets tout au long de leur cycle de vie. Pour plus d'informations, consulter Qu'est-ce que AWS Secrets Manager ? dans le manuel AWS Secrets Manager.

Parameter Store applique les stratégies de paramètre à l'aide d'analyses asynchrones et périodiques. Une fois que vous avez créé une stratégie, vous n'avez pas besoin d'effectuer d'autres actions pour l'appliquer. Parameter Store exécute automatiquement l'action définie par la stratégie en fonction des critères que vous avez spécifiés.

> **Note**
>
> Les stratégies de paramètre sont disponibles pour les paramètres qui utilisent le niveau de paramètres avancés. Pour plus d'informations, consultez la section Gestion des niveaux de paramètres (p. 228).

Une stratégie de paramètre est une séquence JSON, comme illustré dans le tableau suivant. Vous pouvez attribuer une stratégie lorsque vous créez un nouveau paramètre avancé ; vous pouvez aussi appliquer une stratégie en mettant à jour un paramètre. Parameter Store prend en charge les types suivants de stratégies de paramètre.

Policy	Details (Détails)	Exemples
Expiration	Cette stratégie supprime le paramètre. Vous pouvez spécifier une date et une heure spécifiques en utilisant le format `ISO_INSTANT` ou le format `ISO_OFFSET_DATE_TIME`. Pour modifier la date de suppression du paramètre, vous devez mettre à jour la stratégie. La mise à jour d'un paramètre n'affecte pas la date ou l'heure d'expiration de la stratégie qui lui	`{ "Type":"Expiration", "Version":"1.0", "Attributes":{ "Timestamp":"2018-12-02T21:34:33.000Z" } }`

Policy	Details (Détails)	Exemples
	est associée. Lorsque la date et l'heure d'expiration sont atteintes, Parameter Store supprime le paramètre. Note Cet exemple utilise le format `ISO_INSTANT`. Vous pouvez également spécifier une date et une heure en utilisant le format `ISO_OFFSET_DATE_TIME`. Voici un exemple : `2019-11-01T22:13:48.87+10:30:00`.	
ExpirationNotification	Cette stratégie déclenche dans Amazon EventBridge un événement qui vous informe de l'imminence de l'expiration. Cette stratégie vous permet de recevoir des notifications avant la date d'expiration, exprimées en unités de jours ou d'heures.	`{ "Type":"ExpirationNotification", "Version":"1.0", "Attributes":{ "Before":"15", "Unit":"Days" } }`
NoChangeNotification	Cette stratégie déclenche un événement dans EventBridge si un paramètre n'a pas été modifié dans un délai donné. Ce type de stratégie est utile quand, par exemple, un mot de passe doit être modifié dans un délai donné. Cette stratégie détermine quand envoyer une notification en lisant l'attribut `LastModifiedTime` du paramètre. Si vous modifiez un paramètre, le système réinitialise la période de notification en fonction de la nouvelle valeur de `LastModifiedTime`.	`{ "Type":"NoChangeNotification", "Version":"1.0", "Attributes":{ "After":"20", "Unit":"Days" } }`

Vous pouvez attribuer plusieurs stratégies à un paramètre. Par exemple, vous pouvez attribuer `Expiration` des `ExpirationNotification` stratégies et afin que le système déclenche un EventBridge événement vous informant de l'imminence de la suppression d'un paramètre. Vous pouvez attribuer un maximum de dix (10) stratégies à un paramètre.

L'exemple suivant montre une demande dPutParameter'API qui attribue quatre stratégies à un nouveau `SecureString` paramètre nommé `ProdDB3` .

```
PutParameterRequest
{
   "Name":"ProdDB3",
   "Description":"Parameter with policies",
```

```
   "Value":"P@ssW*rd21",
   "Type":"SecureString",
   "Overwrite":"True",
   "Policies":[
      {
         "Type":"Expiration",
         "Version":"1.0",
         "Attributes":{
            "Timestamp":"2018-12-02T21:34:33.000Z"
         }
      },
      {
         "Type":"ExpirationNotification",
         "Version":"1.0",
         "Attributes":{
            "Before":"30",
            "Unit":"Days"
         }
      },
      {
         "Type":"ExpirationNotification",
         "Version":"1.0",
         "Attributes":{
            "Before":"15",
            "Unit":"Days"
         }
      },
      {
         "Type":"NoChangeNotification",
         "Version":"1.0",
         "Attributes":{
            "After":"20",
            "Unit":"Days"
         }
      }
   ]
}
```

Ajout de stratégies à un paramètre existant

Cette section inclut des informations sur la façon d'ajouter des stratégies à un paramètre à l'aide de la console AWS Systems Manager, de l'AWS CLI et des Outils AWS pour Windows PowerShell. Pour plus d'informations sur la création d'un nouveau paramètre incluant des stratégies, consultez Création de paramètres Systems Manager (p. 243).

Rubriques

- Ajouter des stratégies à un paramètre existant (console) (p. 275)
- Ajouter des stratégies à un paramètre existant (AWS CLI) (p. 276)
- Ajouter des stratégies à un paramètre existant avec Outils pour Windows PowerShell (p. 278)

Ajouter des stratégies à un paramètre existant (console)

Utilisez la procédure suivante pour ajouter des stratégies à un paramètre en utilisant la console Systems Manager.

Pour ajouter des stratégies à un paramètre existant

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Parameter Store.

-ou-

Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Parameter Store.

3. Choisissez l'option en regard du paramètre que vous souhaitez mettre à jour afin d'inclure les stratégies, puis choisissez Edit (Modifier).
4. Sélectionnez Advanced.
5. (Facultatif) Dans la section Parameter policies (Stratégies de paramètre), choisissez Enabled (Activé). Vous pouvez spécifier une date d'expiration et une ou plusieurs stratégies de notification pour ce paramètre.
6. Sélectionnez Save Changes (Enregistrer les modifications).

Important

- Parameter Store conserve les stratégies associées à un paramètre jusqu'à ce que vous les remplaciez par de nouvelles stratégies ou que vous les supprimiez.
- Pour supprimer toutes les stratégies d'un paramètre existant, modifiez le paramètre et appliquez une stratégie vide en utilisant des crochets et des accolades, comme suit : `[{}]`
- Si vous ajoutez une nouvelle stratégie à un paramètre qui en possède déjà, Systems Manager remplace les stratégies déjà associées au paramètre. Les stratégies existantes sont supprimées. Si vous souhaitez ajouter une nouvelle stratégie à un paramètre qui en possède déjà une ou plusieurs, vous devez copier et coller les stratégies d'origine, saisir la nouvelle stratégie, puis enregistrer vos modifications.

Ajouter des stratégies à un paramètre existant (AWS CLI)

Utilisez la procédure suivante pour ajouter des stratégies à un paramètre existant à l'aide de l'AWS CLI.

Pour ajouter des stratégies à un paramètre existant

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).
2. Exécutez la commande suivante pour ajouter des stratégies à un paramètre existant.

   Linux

```
aws ssm put-parameter
    --name "parameter-name" \
    --value 'parameter-value' \
    --type parameter-type \
    --overwrite \
    --policies "[{policies-enclosed-in-brackets-and-curly-braces}]"
```

   Windows

```
aws ssm put-parameter
    --name "parameter-name" ^
    --value 'parameter-value' ^
    --type parameter-type ^
    --overwrite ^
    --policies "[{policies-enclosed-in-brackets-and-curly-braces}]"
```

Voici un exemple incluant une stratégie d'expiration qui supprime le paramètre au bout de 15 jours. L'exemple inclut également une stratégie de notification qui génère un EventBridge événement cinq (5) jours avant la suppression du paramètre. Enfin, il inclut une stratégie `NoChangeNotification` si aucune modification n'est apportée à ce paramètre au bout de 60 jours. L'exemple utilise un nom brouillé (`3l3vat3131`) comme mot de passe et une clé principale client (CMK) AWS Key Management Service (AWS KMS). Pour plus d'informations sur CMKs , consultez AWS Key Management Service Concepts dans le AWS Key Management Service Developer Guide .

Linux

```
aws ssm put-parameter \
    --name "/Finance/Payroll/3l3vat3131" \
    --value "P@sSwW)rd" \
    --type "SecureString" \
    --overwrite \
    --policies "[{\"Type\":\"Expiration\",\"Version\":\"1.0\",\"Attributes\":
{\"Timestamp\":\"2020-05-13T00:00:00.000Z\"}},{\"Type\":\"ExpirationNotification
\",\"Version\":\"1.0\",\"Attributes\":{\"Before\":\"5\",\"Unit\":\"Days\"}},{\"Type
\":\"NoChangeNotification\",\"Version\":\"1.0\",\"Attributes\":{\"After\":\"60\",
\"Unit\":\"Days\"}}]"
```

Windows

```
aws ssm put-parameter ^
    --name "/Finance/Payroll/3l3vat3131" ^
    --value "P@sSwW)rd" ^
    --type "SecureString" ^
    --overwrite ^
    --policies "[{\"Type\":\"Expiration\",\"Version\":\"1.0\",\"Attributes\":
{\"Timestamp\":\"2020-05-13T00:00:00.000Z\"}},{\"Type\":\"ExpirationNotification
\",\"Version\":\"1.0\",\"Attributes\":{\"Before\":\"5\",\"Unit\":\"Days\"}},{\"Type
\":\"NoChangeNotification\",\"Version\":\"1.0\",\"Attributes\":{\"After\":\"60\",
\"Unit\":\"Days\"}}]"
```

3. Exécutez la commande suivante pour vérifier les détails du paramètre.

   Linux

```
aws ssm describe-parameters  \
    --parameter-filters "Key=Name,Values=parameter-name"
```

   Windows

```
aws ssm describe-parameters  ^
    --parameter-filters "Key=Name,Values=parameter-name"
```

Important

- Parameter Store conserve les stratégies d'un paramètre jusqu'à ce que vous les remplaciez par de nouvelles stratégies ou que vous les supprimiez.
- Pour supprimer toutes les stratégies d'un paramètre existant, modifiez le paramètre et appliquez une stratégie vide en l'entourant de crochets et d'accolades. Exemple :

  Linux

```
aws ssm put-parameter \
```

```
    --name parameter-name \
    --type parameter-type  \
    --value 'parameter-value' \
    --policies "[{}]"
```

Windows

```
aws ssm put-parameter ^
    --name parameter-name ^
    --type parameter-type  ^
    --value 'parameter-value' ^
    --policies "[{}]"
```

- Si vous ajoutez une nouvelle stratégie à un paramètre qui en possède déjà, Systems Manager remplace les stratégies déjà associées au paramètre. Les stratégies existantes sont supprimées. Si vous souhaitez ajouter une nouvelle stratégie à un paramètre qui en possède déjà une ou plusieurs, vous devez copier et coller les stratégies d'origine, saisir la nouvelle stratégie, puis enregistrer vos modifications.

Ajouter des stratégies à un paramètre existant avec Outils pour Windows PowerShell

Utilisez la procédure suivante pour ajouter des stratégies à un paramètre existant à l'aide des Outils pour Windows PowerShell.

Pour ajouter des stratégies à un paramètre existant

1. Ouvrez les Outils AWS pour Windows PowerShell et exécutez la commande suivante pour spécifier vos informations d'identification. Vous devez disposer de privilèges administrateur dans Amazon EC2 ou vous devez bénéficier de l'autorisation appropriée dans IAM. Pour plus d'informations, consultez la section Prérequis d'Systems Manager (p. 15).

```
Set-AWSCredentials -AccessKey access-key-name -SecretKey secret-key-name
```

2. Exécutez la commande suivante pour définir la région pour votre PowerShell session. L'exemple utilise la région Région USA Est (Ohio) (us-east-2).

```
Set-DefaultAWSRegion -Region us-east-2
```

3. Exécutez la commande suivante pour ajouter des stratégies à un paramètre existant.

```
Write-SSMParameter -Name "parameter-name" -Value "parameter-value" -Type "parameter-
type" -Policies "[{policies-enclosed-in-brackets-and-curly-braces}]" -Overwrite
```

   Voici un exemple incluant une stratégie d'expiration qui supprime le paramètre à minuit (GMT) le 13 mai 2020. L'exemple inclut également une stratégie de notification qui génère un EventBridge événement cinq (5) jours avant la suppression du paramètre. Enfin, il inclut une stratégie `NoChangeNotification` si aucune modification n'est apportée à ce paramètre au bout de 60 jours. L'exemple utilise un nom brouillé (`3l3vat3131`) comme mot de passe et une clé principale client (CMK) AWS.

```
Write-SSMParameter -Name "/Finance/Payroll/3l3vat3131" -Value "P@sSwW)rd" -Type
 "SecureString" -Policies "[{\"Type\":\"Expiration\",\"Version\":\"1.0\",\"Attributes
\":{\"Timestamp\":\"2018-05-13T00:00:00.000Z\"}},{\"Type\":\"ExpirationNotification
\",\"Version\":\"1.0\",\"Attributes\":{\"Before\":\"5\",\"Unit\":\"Days\"}},{\"Type\":
\"NoChangeNotification\",\"Version\":\"1.0\",\"Attributes\":{\"After\":\"60\",\"Unit\":
\"Days\"}}]" -Overwrite
```

4. Exécutez la commande suivante pour vérifier les détails du paramètre.

```
(Get-SSMParameterValue -Name "parameter-name-you-specified").Parameters
```

Important

- Parameter Store conserve les stratégies associées à un paramètre jusqu'à ce que vous les remplaciez par de nouvelles stratégies ou que vous les supprimiez.
- Pour supprimer toutes les stratégies d'un paramètre existant, modifiez le paramètre et appliquez une stratégie vide en l'entourant de crochets et d'accolades. Exemple :

```
Write-SSMParameter -Name "parameter-name" -Value "parameter-value" -Type
 "parameter-type" -Policies "[{}]"
```

- Si vous ajoutez une nouvelle stratégie à un paramètre qui en possède déjà, Systems Manager remplace les stratégies déjà associées au paramètre. Les stratégies existantes sont supprimées. Si vous souhaitez ajouter une nouvelle stratégie à un paramètre qui en possède déjà une ou plusieurs, vous devez copier et coller les stratégies d'origine, saisir la nouvelle stratégie, puis enregistrer vos modifications.

Recherche de paramètres Systems Manager

Lorsque vous avez un grand nombre de paramètres dans votre compte, il peut être difficile de trouver des informations sur un ou plusieurs paramètres à la fois. Dans ce cas, vous pouvez utiliser des outils de filtre pour rechercher ceux sur lesquels vous avez besoin d'informations, en fonction des critères de recherche que vous spécifiez. Pour rechercher des paramètres, vous pouvez utiliser la Systems Manager console AWS CLI , l' Outils AWS pour PowerShell , IDescribeParameters' ou l'API .

Pour de plus amples informations sur l'utilisation de la console Systems Manager et de l'AWS CLI pour afficher des informations sur les paramètres de votre compte à l'aide des options de recherche ou de filtrage, veuillez consulter les rubriques suivantes.

Rubriques

- Rechercher un paramètre (console) (p. 279)
- Rechercher un paramètre (AWS CLI) (p. 280)

Rechercher un paramètre (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Parameter Store.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Parameter Store.
3. Cliquez dans la zone de recherche et choisissez le mode de recherche. Par exemple, `Type` ou `Name`.
4. Fournissez des informations sur le type de recherche que vous avez sélectionné. Exemple :

   - Si vous effectuez une recherche par `Type`, choisissez dans `String`, `StringList`, ou `SecureString`.
   - Si vous effectuez une recherche par `Name`, choisissez `contains`, `equals`, ou `begins-with`, puis saisissez tout ou partie d'un nom de paramètre.

Note

Dans la console , le type de recherche par défaut pour `Name` est `contains`.

5. Appuyez sur Enter.

La liste des paramètres est mise à jour avec les résultats de votre recherche.

Rechercher un paramètre (AWS CLI)

Utilisez la commande `describe-parameters` pour afficher des informations sur un ou plusieurs paramètres dans l'AWS CLI.

Les exemples suivants illustrent différentes options que vous pouvez utiliser pour afficher des informations sur les paramètres de votre compte AWS. Pour de plus amples informations sur ces options, veuillez consulter describe-parameters dans le AWS Command Line Interface Guide de l'utilisateur.

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Remplacez les exemples de valeurs dans les commandes suivantes par des valeurs reflétant les paramètres créés dans votre compte.

   Linux

   ```
   aws ssm describe-parameters \
       --parameter-filters "Key=Name,Values=MyParameterName"
   ```

   Windows

   ```
   aws ssm describe-parameters \
       --parameter-filters "Key=Name,Values=MyParameterName"
   ```

   Note

   Pour `describe-parameters` , le type de recherche par défaut pour `Name` est `Equals` . Dans vos filtres de paramètres, spécifier `"Key=Name,Values=MyParameterName"` est identique à spécifier `"Key=Name,Option=Equals,Values=MyParameterName"` .

```
aws ssm describe-parameters --parameter-filters
 "Key=Name,Option=Contains,Values=Product"
```

```
aws ssm describe-parameters --parameter-filters "Key=Type,Values=String"
```

```
aws ssm describe-parameters --parameter-filters "Key=Path,Values=/Production/West"
```

```
aws ssm describe-parameters --parameter-filters "Key=Tier,Values=Standard"
```

```
aws ssm describe-parameters --parameter-filters "Key=tag:tag-key,Values=tag-value"
```

```
aws ssm describe-parameters --parameter-filters "Key=KeyId,Values=key-id"
```

Note

Dans le dernier exemple, *key-id* représente l'ID d'une clé AWS Key Management Service (AWS KMS) utilisée pour chiffrer un `SecureString` paramètre créé dans votre compte. Vous pouvez également entrer **`alias/aws/ssm`** pour utiliser la clé AWS KMS par défaut de votre compte. Pour plus d'informations, consultez la section Créer un paramètre SecureString (AWS CLI) (p. 251).

Si elle aboutit, la commande renvoie un résultat semblable au suivant :

```
{
    "Parameters": [
        {
            "Name": "/Production/West/Manager",
            "Type": "String",
            "LastModifiedDate": 1573438580.703,
            "LastModifiedUser": "arn:aws:iam::111122223333:user/Mateo.Jackson",
            "Version": 1,
            "Tier": "Standard",
            "Policies": []
        },
        {
            "Name": "/Production/West/TeamLead",
            "Type": "String",
            "LastModifiedDate": 1572363610.175,
            "LastModifiedUser": "arn:aws:iam::111122223333:user/Mateo.Jackson",
            "Version": 1,
            "Tier": "Standard",
            "Policies": []
        },
        {
            "Name": "/Production/West/HR",
            "Type": "String",
            "LastModifiedDate": 1572363680.503,
            "LastModifiedUser": "arn:aws:iam::111122223333:user/Mateo.Jackson",
            "Version": 1,
            "Tier": "Standard",
            "Policies": []
        }
    ]
}
```

Utilisation des paramètres à l'aide Fonctionnalité Exécuter la commande des commandes de l'

Vous pouvez utiliser des paramètres dans Fonctionnalité Exécuter la commande . Pour plus d'informations, consultez .Fonctionnalité Exécuter la commandeExécuter des commandes avec Systems Manager Fonctionnalité Exécuter la commande (p. 946)

Exécution d'un paramètre String (console)

La procédure suivante vous guide à travers le processus d'exécution d'une commande qui utilise un `String` paramètre.

Pour exécuter un paramètre String à l'aide deParameter Store

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fonctionnalité Exécuter la commande.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Fonctionnalité Exécuter la commande.
3. Choisissez Exécuter une commande.
4. Dans la liste Command document (Document de commande), choisissez `AWS-RunPowerShellScript` (Windows) ou `AWS-RunShellScript` (Linux).
5. Pour Command parameters (Paramètres de commande), entrez .**`echo {{ssm:`*`parameter-name`*`}}`** Par exemple : **`echo {{ssm:/Test/helloWorld}}`**.
6. Dans la section Cibles, identifiez les instances sur lesquelles vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant manuellement les instances ou en indiquant un groupe de ressources.

   Note

   Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.
7. Pour Autres paramètres :
   - Pour Comment (Commentaire), saisissez des informations à propos de cette commande.
   - Pour Délai (secondes), précisez le nombre de secondes durant lesquelles le système doit attendre avant de mettre en échec l'exécution de la commande globale.
8. (Facultatif) Pour Contrôle du débit :
   - Pour Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage d'instances sur lesquelles exécuter simultanément la commande.

     Note

     Si vous avez sélectionné des cibles en indiquant des balises appliquées à des instances gérées ou en indiquant des groupes de ressources AWS, et que vous n'êtes pas sûr du nombre d'instances ciblées, limitez le nombre d'instances pouvant exécuter simultanément le document en spécifiant un pourcentage.
   - Pour Seuil d'erreur, spécifiez quand arrêter l'exécution de la commande sur d'autres instances après son échec sur un nombre ou un pourcentage d'instances. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les instances sur lesquelles la commande est en cours de traitement peuvent également envoyer des erreurs.
9. (Facultatif) Pour Output options (Options de sortie), pour enregistrer la sortie de la commande dans un fichier, cochez la case Write command output to an S3 bucket (Écrire la sortie de commande vers un compartiment S3). Entrez les noms de compartiment et de préfixe (dossier) dans les zones.

   Note

   Les autorisations S3 qui accordent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour de plus amples informations, veuillez consulter Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). En outre, si le compartiment S3 spécifié se trouve dans un compte AWS différent, assurez-vous que le profil d'instance associé à l'instance dispose des autorisations nécessaires pour écrire dans ce compartiment.

10. Dans la section SNS notifications (Notifications SNS), si vous souhaitez que des notifications sur le statut de l'exécution de la commande soient envoyées, cochez la case Enable SNS notifications (Activer les notifications SNS).

    Pour plus d'informations sur la configuration des notifications Amazon SNS pour la fonctionnalité Exécuter la commande, consultez Surveillance des changements de statut Systems Manager à l'aide de notifications Amazon SNS (p. 1344).
11. Cliquez sur Exécuter.
12. Sur la page Command ID (ID de commande) dans la zone Targets and outputs (Cibles et sorties) sélectionnez le bouton en regard de l'ID d'une instance dans laquelle vous avez exécuté la commande, puis choisissez View output (Afficher la sortie). Vérifiez que la sortie de la commande est la valeur que vous avez fournie pour le paramètre, par exemple, **`This is my first parameter`**.

Exécution d'un paramètre (interface de ligne de commande)

L'exemple de commande suivant inclut un Systems Manager paramètre nommé `DNS-IP` . La valeur de ce paramètre est simplement l'adresse IP d'une instance. Cet exemple utilise une commande de l'AWS CLI pour reprendre la valeur du paramètre.

Linux

```
aws ssm send-command \
    --document-name "AWS-RunPowerShellScript" \
    --document-version "1" \
    --targets "Key=instanceids,Values=i-02573cafcfEXAMPLE" \
    --parameters "commands='echo {{ssm:DNS-IP}}'" \
    --timeout-seconds 600 \
    --max-concurrency "50" \
    --max-errors "0" \
    --region us-east-2
```

Windows

```
aws ssm send-command ^
    --document-name "AWS-RunPowerShellScript" ^
    --document-version "1" ^
    --targets "Key=instanceids,Values=i-02573cafcfEXAMPLE" ^
    --parameters "commands='echo {{ssm:DNS-IP}}'" ^
    --timeout-seconds 600 ^
    --max-concurrency "50" ^
    --max-errors "0" ^
    --region us-east-2
```

L'exemple de commande suivant utilise un `SecureString` paramètre nommé SecurePassword . La commande `commands=['$secure = (Get-SSMParameterValue -Names SecurePassword -WithDecryption $True).Parameters[0].Value','net user administrator $secure']` récupère et déchiffre la valeur du `SecureString` paramètre, puis réinitialise le mot de passe administrateur local sans avoir à le transmettre en texte clair.

Linux

```
aws ssm send-command \
        --document-name "AWS-RunPowerShellScript" \
        --document-version "1" \
        --targets "Key=instanceids,Values=i-02573cafcfEXAMPLE" \
        --parameters "commands=['$secure = (Get-SSMParameterValue -Names SecurePassword
 -WithDecryption $True).Parameters[0].Value','net user administrator $secure']" \
```

```
        --timeout-seconds 600 \
        --max-concurrency "50" \
        --max-errors "0" \
        --region us-east-2
```

Windows

```
aws ssm send-command ^
        --document-name "AWS-RunPowerShellScript" ^
        --document-version "1" ^
        --targets "Key=instanceids,Values=i-02573cafcfEXAMPLE" ^
        --parameters "commands=['$secure = (Get-SSMParameterValue -Names SecurePassword
 -WithDecryption $True).Parameters[0].Value','net user administrator $secure']" ^
        --timeout-seconds 600 ^
        --max-concurrency "50" ^
        --max-errors "0" ^
        --region us-east-2
```

Vous pouvez aussi référencer des paramètres Systems Manager dans la section Paramètres d'un document SSM, comme le montre l'exemple suivant.

```
{
   "schemaVersion":"2.0",
   "description":"Sample version 2.0 document v2",
   "parameters":{
      "commands" : {
        "type": "StringList",
        "default": ["{{ssm:parameter-name}}"]
      }
    },
    "mainSteps":[
       {
          "action":"aws:runShellScript",
          "name":"runShellScript",
          "inputs":{
             "runCommand": "{{commands}}"
          }
       }
    ]
}
```

Évitez de confondre la syntaxe similaire pour les paramètres locaux utilisés dans la section `runtimeConfig` des documents SSM avec les paramètres Parameter Store. Un paramètre local est différent d'un paramètre Systems Manager. Vous pouvez distinguer les paramètres locaux des paramètres Systems Manager par l'absence du préfixe `ssm:`.

```
"runtimeConfig":{
        "aws:runShellScript":{
            "properties":[
                {
                    "id":"0.aws:runShellScript",
                    "runCommand":"{{ commands }}",
                    "workingDirectory":"{{ workingDirectory }}",
                    "timeoutSeconds":"{{ executionTimeout }}"
```

Note

Les documents SSM ne prennent actuellement pas en charge les références aux paramètres `SecureString`. Cela signifie que pour utiliser `SecureString` des paramètres avec, par exemple, Fonctionnalité Exécuter la commande , vous devez récupérer la valeur du paramètre

avant de la transmettre à Fonctionnalité Exécuter la commande , comme illustré dans les exemples suivants.

Linux

```
value=$(aws ssm get-parameters --names parameter-name --with-decryption)
```

```
aws ssm send-command \
    --name AWS-JoinDomain \
    --parameters password=$value \
    --instance-id instance-id
```

Windows

```
aws ssm send-command ^
    --name AWS-JoinDomain ^
    --parameters password=$value ^
    --instance-id instance-id
```

Outils pour Windows PowerShell

```
$secure = (Get-SSMParameterValue -Names parameter-name -WithDecryption
 $True).Parameters[0].Value | ConvertTo-SecureString -AsPlainText -Force
```

```
$cred = New-Object System.Management.Automation.PSCredential -argumentlist user-
name,$secure
```

Utilisation des hiérarchies de paramètres

La gestion de douzaines ou de centaines de paramètres comme une liste simple est chronophage et propice aux erreurs. Il peut également être difficile d'identifier le bon paramètre pour une tâche. Cela signifie que vous utilisez peut-être par erreur le mauvais paramètre ou que vous créez peut-être plusieurs paramètres qui utilisent les mêmes données de configuration.

Vous pouvez utiliser les hiérarchies des paramètres pour vous aider à organiser et à gérer des paramètres . Une hiérarchie est un nom de paramètre qui comporte un chemin que vous définissez en utilisant des barres obliques (/).

Rubriques

- Exemples de hiérarchie de paramètres (p. 285)
- Interrogation de paramètres dans une hiérarchie (p. 286)
- Restriction de l'accès aux actions de l'API Parameter Store (p. 287)
- Gérer les paramètres en utilisant les hiérarchies (AWS CLI) (p. 287)

Exemples de hiérarchie de paramètres

L'exemple suivant utilise trois niveaux de hiérarchie dans le nom pour identifier ce qui suit :

`/Environment/Type of computer/Application/Data`

`/Dev/DBServer/MySQL/db-string13`

Vous pouvez créer une hiérarchie avec un maximum de 15 niveaux. Nous vous suggérons de créer des hiérarchies qui reflètent une structure hiérarchique existante dans votre environnement, comme indiqué dans les exemples suivants :

- Votre environnement d'intégration continue et de livraison continue (flux de travail IC/LC)

  `/Dev/DBServer/MySQL/db-string`

  `/Staging/DBServer/MySQL/db-string`

  `/Prod/DBServer/MySQL/db-string`
- Vos applications qui utilisent des conteneurs

```
/MyApp/.NET/Libraries/my-password
```

- L'organisation de votre entreprise

  `/Finance/Accountants/UserList`

  `/Finance/Analysts/UserList`

  `/HR/Employees/EU/UserList`

Les hiérarchies des paramètres standardisent la façon dont vous créez les paramètres et facilitent la gestion des paramètres dans le temps. Une hiérarchie de paramètres peut aussi vous aider à identifier le bon paramètre pour une tâche de configuration. Cela vous évite de créer plusieurs paramètres avec les mêmes données de configuration.

Vous pouvez créer une hiérarchie qui vous permet de partager des paramètres entre différents environnements, comme illustré dans les exemples suivants qui utilisent des mots de passe dans l'environnement de développement et de transit.

`/DevTest/MyApp/database/my-password`

Vous pouvez ensuite créer un mot de passe unique pour votre environnement de production, comme illustré dans l'exemple suivant :

`/prod/MyApp/database/my-password`

Vous n'avez pas besoin de spécifier une hiérarchie de paramètres. Vous pouvez créer des paramètres au niveau un. Ils sont appelés racine Paramètres . Pour assurer la compatibilité descendante, tous les paramètres créés dans Parameter Store avant la mise à disposition des hiérarchies sont des paramètres racine. Le système traite les deux paramètres suivants comme des paramètres racines.

`/parameter-name`

`parameter-name`

Interrogation de paramètres dans une hiérarchie

Un autre avantage de l'utilisation des hiérarchies est la possibilité d'interroger tous les paramètres d'une hiérarchie à l'aide de l'action GetParametersByPath d'API. Pour exemple, si vous exécutez la commande suivante à partir de l'AWS CLI, le système renvoie tous les paramètres au niveau IIS.

```
aws ssm get-parameters-by-path --path /Dev/Web/IIS
```

Pour afficher les paramètres `SecureString` déchiffrés dans une hiérarchie, vous spécifiez le chemin et le paramètre `--with-decryption`, comme illustré dans l'exemple suivant.

```
aws ssm get-parameters-by-path --path /Prod/ERP/SAP --with-decryption
```

Restriction de l'accès aux actions de l'API Parameter Store

À l'aide des stratégies AWS Identity and Access Management (IAM), vous pouvez autoriser ou restreindre l'accès des utilisateurs aux actions et au contenu de l'API Parameter Store.

Dans l'exemple de stratégie suivant, les utilisateurs sont d'abord autorisés à accéder pour exécuter l'action d'API `PutParameter` sur tous les paramètres du compte AWS 123456789012 dans la région Région USA Est (Ohio) (us-east-2). Mais les utilisateurs n'ont ensuite pas le droit de modifier les valeurs des paramètres existants car l'option `Overwrite` est explicitement refusée pour l'action `PutParameter`. En d'autres termes, les utilisateurs auxquels cette stratégie est affectée peuvent créer des paramètres, mais ne peuvent pas modifier les paramètres existants.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:PutParameter"
            ],
            "Resource": "arn:aws:ssm:us-east-2:123456789012:parameter/*"
        },
        {
            "Effect": "Deny",
            "Action": [
                "ssm:PutParameter"
            ],
            "Condition": {
                "StringEquals": {
                    "ssm:Overwrite": [
                        "true"
                    ]
                }
            },
            "Resource": "arn:aws:ssm:us-east-2:123456789012:parameter/*"
        }
    ]
}
```

Gérer les paramètres en utilisant les hiérarchies (AWS CLI)

Cette procédure montre comment utiliser les paramètres et les hiérarchies de paramètres à l'aide de l' AWS CLI .

Pour gérer les paramètres en utilisant les hiérarchies

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).
2. Exécutez la commande suivante pour créer un paramètre utilisant le paramètre `allowedPattern` et le type de paramètre `String`. Le modèle autorisé dans cet exemple indique que la valeur du paramètre doit comporter entre 1 et 4 chiffres.

   Linux

   ```
   aws ssm put-parameter \
       --name "/MyService/Test/MaxConnections" \
       --value 100 --allowed-pattern "\d{1,4}" \
       --type String
   ```

Windows

```
aws ssm put-parameter ^
    --name "/MyService/Test/MaxConnections" ^
    --value 100 --allowed-pattern "\d{1,4}" ^
    --type String
```

La commande renvoie le numéro de version du paramètre.

3. Exécutez la commande suivante pour essayer de remplacer la paramètre que vous venez de créer avec une nouvelle valeur.

   Linux

```
aws ssm put-parameter \
    --name "/MyService/Test/MaxConnections" \
    --value 10,000 \
    --type String \
    --overwrite
```

   Windows

```
aws ssm put-parameter ^
    --name "/MyService/Test/MaxConnections" ^
    --value 10,000 ^
    --type String ^
    --overwrite
```

   Le système renvoie l'erreur suivante, car la nouvelle valeur ne répond pas aux exigences du modèle autorisé que vous avez spécifié à l'étape précédente.

```
An error occurred (ParameterPatternMismatchException) when calling the PutParameter
 operation: Parameter value, cannot be validated against allowedPattern: \d{1,4}
```

4. Exécutez la commande suivante pour créer un paramètre `SecureString` utilisant une clé CMK gérée par AWS. Le modèle autorisé dans cet exemple indique que l'utilisateur peut spécifique n'importe quel caractère, et la valeur doit comporter entre 8 et 20 caractères.

   Linux

```
aws ssm put-parameter \
    --name "/MyService/Test/my-password" \
    --value "p#sW*rd33" \
    --allowed-pattern ".{8,20}" \
    --type SecureString
```

   Windows

```
aws ssm put-parameter ^
    --name "/MyService/Test/my-password" ^
    --value "p#sW*rd33" ^
    --allowed-pattern ".{8,20}" ^
    --type SecureString
```

5. Exécutez les commandes suivantes pour créer plus de paramètres qui utilisent la structure de hiérarchie à partir de l'étape précédente.

Linux

```
aws ssm put-parameter \
    --name "/MyService/Test/DBname" \
    --value "SQLDevDb" \
    --type String
```

```
aws ssm put-parameter \
    --name "/MyService/Test/user" \
    --value "SA" \
    --type String
```

```
aws ssm put-parameter \
    --name "/MyService/Test/userType" \
    --value "SQLuser" \
    --type String
```

Windows

```
aws ssm put-parameter ^
    --name "/MyService/Test/DBname" ^
    --value "SQLDevDb" ^
    --type String
```

```
aws ssm put-parameter ^
    --name "/MyService/Test/user" ^
    --value "SA" ^
    --type String
```

```
aws ssm put-parameter ^
    --name "/MyService/Test/userType" ^
    --value "SQLuser" ^
    --type String
```

6. Exécutez la commande suivante pour obtenir la valeur de deux paramètres.

   Linux

```
aws ssm get-parameters \
    --names "/MyService/Test/user" "/MyService/Test/userType"
```

   Windows

```
aws ssm get-parameters ^
    --names "/MyService/Test/user" "/MyService/Test/userType"
```

7. Exécutez la commande suivante pour interroger tous les paramètres à un seul niveau.

   Linux

```
aws ssm get-parameters-by-path \
    --path "/MyService/Test"
```

Windows

```
aws ssm get-parameters-by-path ^
    --path "/MyService/Test"
```

8. Exécutez la commande suivante pour supprimer deux paramètres

   Linux

```
aws ssm delete-parameters \
    --names "/IADRegion/Dev/user" "/IADRegion/Dev/userType"
```

   Windows

```
aws ssm delete-parameters ^
    --names "/IADRegion/Dev/user" "/IADRegion/Dev/userType"
```

Utilisation des étiquettes de paramètre

Une étiquette de paramètre est un alias défini par l'utilisateur pour vous aider à gérer les différentes versions d'un paramètre. Lorsque vous modifiez un paramètre, Systems Manager enregistre automatiquement une nouvelle version et incrémente le numéro de version d'une unité. Une étiquette peut vous aider à vous souvenir de l'objectif d'une version de paramètre lorsqu'il existe plusieurs versions.

Par exemple, supposons que vous avez un paramètre appelé /MyApp/DB/ConnectionString. La valeur du paramètre est une chaîne de connexion à un MySQL serveur d'une base de données locale dans un environnement de test. Une fois que vous avez terminé la mise à jour de l'application, vous souhaitez que le paramètre utilise une chaîne de connexion pour une base de données de production. Vous modifiez la valeur de /MyApp/DB/ConnectionString. Systems Manager crée automatiquement la version 2 avec la nouvelle chaîne de connexion. Pour vous aider à vous souvenir de l'objectif de chaque version, vous attachez une étiquette à chaque paramètre. Pour la version 1, vous attachez l'étiquette Test et pour la version 2, vous attachez l'étiquette Production.

Vous pouvez déplacer les étiquettes d'une version d'un paramètre à une autre version. Par exemple, si vous créez la version 3 du paramètre /MyApp/DB/ConnectionString avec une chaîne de connexion pour une nouvelle base de données de production, vous pouvez déplacer l'étiquette Production du paramètre 2 vers le paramètre 3.

Les étiquettes de paramètre sont une alternative légère aux balises de paramètre. Votre organisation peut avoir des consignes strictes concernant les balises qui doivent être appliquées aux différentes ressources AWS. En revanche, une étiquette est une simple association de texte pour une version spécifique d'un paramètre.

À l'instar des balises, vous pouvez interroger les paramètres à l'aide des étiquettes. Vous pouvez afficher une liste des versions de paramètres spécifiques qui utilisent toutes la même étiquette si vous interrogez votre ensemble de paramètres à l'aide de l'action GetParametersByPath d'API , comme décrit plus loin dans cette section.

Exigences et restrictions liées aux étiquettes

Les étiquettes de paramètre possèdent les exigences et les restrictions suivantes :

- Une version d'un paramètre peut avoir un maximum de 10 étiquettes.
- Vous ne pouvez pas attacher la même étiquette à différentes versions d'un même paramètre. Par exemple, si la version 1 a l'étiquette Production, vous ne pouvez pas attacher Production à la version 2.

- Vous pouvez déplacer une étiquette d'une version d'un paramètre à une autre.
- Vous ne pouvez pas créer d'étiquette lorsque vous créez un nouveau paramètre. Vous devez attacher une étiquette à une version spécifique d'un paramètre.
- Vous ne pouvez pas supprimer une étiquette de paramètre. Si vous ne souhaitez plus utiliser une étiquette de paramètre, vous devez la déplacer vers une autre version d'un paramètre.
- Une étiquette peut comporter un maximum de 100 caractères.
- Les étiquettes peuvent contenir des lettres (sensibles à la casse), des chiffres, des points (.), des tirets (-) et des traits de soulignement (_).
- Les étiquettes ne peuvent pas commencer par un chiffre, par « aws » ni par « ssm » (non sensible à la casse). Si une étiquette ne répond pas à ces exigences, elle n'est pas attachée à la version du paramètre et le système l'affiche dans la liste `InvalidLabels`.

Rubriques

- Utilisation des étiquettes de paramètre (console) (p. 291)
- Utilisation des étiquettes de paramètre (AWS CLI) (p. 292)

Utilisation des étiquettes de paramètre (console)

Cette section explique comment effectuer les tâches suivantes à l'aide de la console AWS Systems Manager.

- Créer une nouvelle étiquette de paramètre (console) (p. 291)
- Afficher les étiquettes attachées à un paramètre (console) (p. 292)
- Déplacement d'une étiquette de paramètre (console) (p. 292)

Créer une nouvelle étiquette de paramètre (console)

La procédure suivante indique comment attacher une étiquette à une version spécifique d'un paramètre existant à l'aide de la console Systems Manager. Vous ne pouvez pas attacher d'étiquette lorsque vous créez un paramètre.

Pour attacher une étiquette à une version de paramètre à l'aide de la console

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Parameter Store.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Parameter Store.
3. Choisissez le nom d'un paramètre pour ouvrir la page des détails de ce paramètre.
4. Choisissez l'onglet History (Historique).
5. Choisissez la version du paramètre pour laquelle vous souhaitez attacher une étiquette.
6. Choisissez Attach labels (Attacher des étiquettes).
7. Choisissez Add another label (Ajouter une autre étiquette).
8. Dans la zone de texte, entrez l'étiquette. Pour ajouter d'autres étiquettes, choisissez Add another label (Ajouter une autre étiquette). Vous pouvez attacher un maximum de dix étiquettes.
9. Lorsque vous avez terminé d'attacher des étiquettes, choisissez Confirmer.

Afficher les étiquettes attachées à un paramètre (console)

Une version de paramètre peut avoir un maximum de dix étiquettes. La procédure suivante explique comment afficher toutes les étiquettes attachées à une version de paramètre à l'aide de la console Systems Manager.

Pour afficher les étiquettes attachées à une version de paramètre à l'aide de la console

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Parameter Store.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Parameter Store.
3. Choisissez le nom d'un paramètre pour ouvrir la page des détails de ce paramètre.
4. Choisissez l'onglet History (Historique).
5. Localisez la version du paramètre pour laquelle vous souhaitez afficher toutes les étiquettes attachées. La colonne Labels (Étiquettes) indique toutes les étiquettes attachées à la version de paramètre.

Déplacement d'une étiquette de paramètre (console)

Vous ne pouvez pas supprimer une étiquette de paramètre après l'avoir créée. Toutefois, vous pouvez déplacer une étiquette entre les versions d'un paramètre. La procédure suivante explique comment déplacer une étiquette de paramètre vers une autre version du même paramètre à l'aide de la console Systems Manager.

Pour déplacer une étiquette vers une autre version du paramètre à l'aide de la console

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Parameter Store.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Parameter Store.
3. Choisissez le nom d'un paramètre pour ouvrir la page des détails de ce paramètre.
4. Choisissez l'onglet History (Historique).
5. Choisissez la version du paramètre pour laquelle vous souhaitez déplacer l'étiquette.
6. Choisissez Attach labels (Attacher des étiquettes).
7. Choisissez Add another label (Ajouter une autre étiquette).
8. Dans la zone de texte, entrez l'étiquette. La console vous informe du déplacement de l'étiquette.
9. Lorsque vous avez terminé, choisissez Confirmer.

Utilisation des étiquettes de paramètre (AWS CLI)

Cette section explique comment effectuer les tâches suivantes à l'aide de l'AWS CLI.

- Créer une nouvelle étiquette de paramètre (interface de ligne de commande) (p. 293)
- Affichage des étiquettes d'un paramètre (interface de ligne de commande) (p. 294)

- Afficher une liste des paramètres auxquels une étiquette est attribuée (interface de ligne de commande) (p. 295)
- Déplacement d'une étiquette de paramètre (interface de ligne de commande) (p. 296)

Créer une nouvelle étiquette de paramètre (interface de ligne de commande)

La procédure suivante indique comment attacher une étiquette à une version spécifique d'un paramètre existant à l'aide de l'AWS CLI. Vous ne pouvez pas attacher d'étiquette lorsque vous créez un paramètre.

Pour créer une nouvelle étiquette de paramètre

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Exécutez la commande suivante pour afficher la liste des paramètres auxquels vous êtes autorisé à attacher une étiquette.

   > **Note**
   >
   > Les paramètres sont uniquement disponibles dans la région où ils ont été créés. Si vous ne trouvez pas le paramètre auquel vous souhaitez attacher une étiquette, vérifiez votre région.

   ```
   aws ssm describe-parameters
   ```

   Notez le nom d'un paramètre auquel vous souhaitez attacher une étiquette.

3. Exécutez la commande suivante pour afficher toutes les versions du paramètre.

   ```
   aws ssm get-parameter-history --name "parameter-name"
   ```

   Notez la version du paramètre à laquelle vous souhaitez attacher une étiquette.

4. Exécutez la commande suivante pour récupérer des informations sur un paramètre par numéro de version.

   ```
   aws ssm get-parameters --names "parameter-name:version-number"
   ```

   Voici un exemple de .

   ```
   aws ssm get-parameters --names "/Production/SQLConnectionString:3"
   ```

5. Exécutez l'une des commandes suivantes pour attacher une étiquette à une version d'un paramètre. Si vous attachez plusieurs étiquettes, vous devez séparer leurs noms par un espace.

   Attacher une étiquette à la dernière version d'un paramètre

   ```
   aws ssm label-parameter-version --name parameter-name  --labels label-name
   ```

   Attacher une étiquette à une version spécifique d'un paramètre

   ```
   aws ssm label-parameter-version --name parameter-name --parameter-version version-
   number --labels label-name
   ```

   Voici quelques exemples :

```
aws ssm label-parameter-version --name /config/endpoint --labels production east-region
 finance
```

```
aws ssm label-parameter-version --name /config/endpoint --parameter-version 3 --labels
 MySQL-test
```

> **Note**
>
> Si la sortie montre l'étiquette que vous avez créée dans la liste `InvalidLabels`, l'étiquette ne respecte pas les exigences décrites plus haut dans cette rubrique. Passez en revue les exigences et réessayez. Si la liste `InvalidLabels` est vide, votre étiquette a été correctement appliquée à la version du paramètre.

6. Vous pouvez afficher les détails du paramètre en utilisant un numéro de version ou un nom d'étiquette. Exécutez la commande suivante et spécifiez l'étiquette que vous avez créée à l'étape précédente.

```
aws ssm get-parameter --name parameter-name:label-name --with-decryption
```

La commande renvoie des informations semblables aux suivantes :

```
{
    "Parameter": {
        "Version": version-number,
        "Type": "parameter-type",
        "Name": "parameter-name",
        "Value": "parameter-value",
        "Selector": ":label-name"
    }
}
```

> **Note**
>
> Selector dans la sortie correspond au numéro de version ou à l'étiquette que vous avez spécifiée dans le champ de saisie `Name`.

Affichage des étiquettes d'un paramètre (interface de ligne de commande)

Vous pouvez utiliser l'action GetParameterHistory d'API pour afficher l'historique complet et toutes les étiquettes attachées à un paramètre spécifié. Vous pouvez également utiliser l'action GetParametersByPath d'API pour afficher la liste de tous les paramètres auxquels une étiquette spécifique est attribuée.

Pour afficher les étiquettes d'un paramètre à l'aide de l'action GetParameterHistory d'API

1. Exécutez la commande suivante pour afficher la liste des paramètres dont vous pouvez afficher les étiquettes.

   > **Note**
   >
   > Les paramètres sont uniquement disponibles dans la région où ils ont été créés. Si vous ne trouvez pas de paramètre pour lequel vous souhaitez déplacer une étiquette, vérifiez votre région.

```
aws ssm describe-parameters
```

Notez le nom d'un paramètre dont vous souhaitez déplacer une étiquette.

2. Exécutez la commande suivante pour afficher toutes les versions du paramètre.

```
aws ssm get-parameter-history --name parameter-name --with-decryption
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
    "Parameters": [
        {
            "Name": "/Config/endpoint",
            "LastModifiedDate": 1528932105.382,
            "Labels": [
                "Deprecated"
            ],
            "Value": "MyTestService-June-Release.example.com",
            "Version": 1,
            "LastModifiedUser": "arn:aws:iam::123456789012:user/test",
            "Type": "String"
        },
        {
            "Name": "/Config/endpoint",
            "LastModifiedDate": 1528932111.222,
            "Labels": [
                "Current"
            ],
            "Value": "MyTestService-July-Release.example.com",
            "Version": 2,
            "LastModifiedUser": "arn:aws:iam::123456789012:user/test",
            "Type": "String"
        }
    ]
}
```

Afficher une liste des paramètres auxquels une étiquette est attribuée (interface de ligne de commande)

Vous pouvez utiliser l'action GetParametersByPath d'API pour afficher une liste de tous les paramètres dans un chemin d'accès auxquels une étiquette spécifique est attribuée.

Exécutez la commande suivante pour afficher la liste des paramètres dans un chemin d'accès auxquels une étiquette spécifique est attribuée.

```
aws ssm get-parameters-by-path --path parameter-path --parameter-filters
 Key=Label,Values=label-name,Option=Equals --max-results a-number --with-decryption --
recursive
```

Le système renvoie des informations similaires à celles qui suivent. Pour cet exemple, l'utilisateur a recherché dans le chemin d'accès /Config :

```
{
    "Parameters": [
        {
            "Version": 3,
            "Type": "SecureString",
            "Name": "/Config/DBpwd",
            "Value": "MyS@perGr&pass33"
        },
        {
            "Version": 2,
```

```
            "Type": "String",
            "Name": "/Config/DBusername",
            "Value": "TestUserDB"
        },
        {
            "Version": 2,
            "Type": "String",
            "Name": "/Config/endpoint",
            "Value": "MyTestService-July-Release.example.com"
        }
    ]
}
```

Déplacement d'une étiquette de paramètre (interface de ligne de commande)

Vous ne pouvez pas supprimer une étiquette de paramètre après l'avoir créée. Toutefois, vous pouvez déplacer une étiquette entre les versions d'un paramètre. La procédure suivante explique comment déplacer une étiquette de paramètre vers une autre version du même paramètre.

Pour déplacer une étiquette de paramètre

1. Exécutez la commande suivante pour afficher toutes les versions du paramètre.

   ```
   aws ssm get-parameter-history --name "parameter-name"
   ```

   Notez la version du paramètre à laquelle vous souhaitez attacher une étiquette.
2. Exécutez la commande suivante pour affecter une étiquette existante à une autre version d'un paramètre.

   ```
   aws ssm label-parameter-version --name parameter-name --parameter-version version-
   number --labels name-of-existing-label
   ```

   > **Note**
   >
   > Si vous souhaitez déplacer une étiquette existante vers la dernière version d'un paramètre, supprimez `--parameter-version` de la commande.

Utilisation des versions de paramètre

Chaque fois que vous modifiez la valeur d'un paramètre, Parameter Store crée une nouvelle version du paramètre et conserve les versions précédentes. Lorsque vous créez un paramètre, Parameter Store lui affecte la version `1`. Lorsque vous modifiez la valeur du paramètre, Parameter Store itère automatiquement le numéro de version par un. Vous pouvez afficher les détails, y compris les valeurs, de toutes les versions de l'historique d'un paramètre.

Vous pouvez également spécifier la version d'un paramètre à utiliser dans les commandes d'API et les documents SSM ; par exemple : `ssm:MyParameter:3`. Vous pouvez spécifier un nom de paramètre et un numéro de version spécifique dans les appels d'API et les documents SSM. Si vous ne spécifiez pas de numéro de version, le système utilise automatiquement la dernière version.

Vous pouvez également utiliser les versions de paramètre pour savoir combien de fois un paramètre a été modifié sur une période donnée. Les versions de paramètres fournissent également une couche de protection si une valeur de paramètre est accidentellement modifiée.

Vous pouvez créer et gérer jusqu'à 100 versions d'un paramètre. Une fois que vous avez créé 100 versions d'un paramètre, chaque fois que vous créez une nouvelle version, la version la plus ancienne du paramètre est supprimée de l'historique pour faire de la place à la nouvelle version.

L'exception à cette règle est lorsqu'il y a déjà 100 versions de paramètres dans l'historique et qu'une étiquette de paramètre est attribuée à la version la plus ancienne d'un paramètre. Dans ce cas, cette version n'est pas supprimée de l'historique et la demande de création d'une nouvelle version de paramètre échoue. Cette protection a pour but d'empêcher la suppression des versions de paramètres auxquelles des étiquettes critiques leur sont attribuées. Pour continuer à créer de nouveaux paramètres, déplacez d'abord l'étiquette de la version la plus ancienne du paramètre vers une version plus récente pour l'utiliser dans vos opérations. Pour plus d'informations sur le déplacement des étiquettes de paramètres, consultez Déplacement d'une étiquette de paramètre (console) (p. 292) et Déplacement d'une étiquette de paramètre (interface de ligne de commande) (p. 296) .

Les procédures suivantes vous montrent comment modifier un paramètre, puis vérifier qu'une nouvelle version a été créée.

Rubriques

- Créer une nouvelle version d'un paramètre (console) (p. 297)
- Référence d'une version de paramètre (p. 297)

Créer une nouvelle version d'un paramètre (console)

Vous pouvez utiliser la console AWS Systems Manager pour créer une nouvelle version d'un paramètre et afficher l'historique des versions d'un paramètre.

Pour créer une nouvelle version d'un paramètre

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Parameter Store.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Parameter Store.
3. Choisissez le nom d'un paramètre que vous avez créé précédemment. Pour plus d'informations sur la création d'un nouveau paramètre, consultez Création de paramètres Systems Manager (p. 243).
4. Choisissez Edit.
5. Dans la boîte de dialogue Valeur, entrez une nouvelle valeur, puis choisissez Save changes.
6. Choisissez le nom du paramètre que vous venez de mettre à jour. Dans l'onglet Présentation, vérifiez que le numéro de version a été incrémenté de 1, et vérifiez la nouvelle valeur.
7. Pour afficher l'historique de toutes les versions d'un paramètre, choisissez l'onglet Historique .

Référence d'une version de paramètre

Vous pouvez référencer des versions de paramètres spécifiques dans les commandes, les appels d'API et les documents SSM en utilisant le format suivant : ssm : *`parameter-name`*`:`*`version-number`*.

Dans l'exemple suivant, la version 3 du paramètre `golden-ami` est utilisée dans la `run-instances command` EC2.

```
aws ec2 run-instances \
    --image-id resolve:ssm:/golden-ami:3 \
    --count 1 \
    --instance-type t2.micro \
    --key-name my-key-pair \
    --security-groups my-security-group
```

> **Note**
>
> Actuellement, l'utilisation de `resolve` et d'une valeur de paramètre ne fonctionne qu'avec l'option `--image-id` et un paramètre qui contient une AMI (Amazon Machine Image) comme valeur. Pour plus d'informations, consultez la section Prise en charge des paramètres natifs pour Amazon Machine ImageIDs (p. 298).

Voici un exemple pour spécifier la version 2 d'un paramètre nommé `MyRunCommandParameter` dans un document SSM.

```
{
    "schemaVersion": "2.2",
    "description": "Run a shell script or specify the commands to run.",
    "parameters": {
        "commands": {
            "type": "String",
            "description": "(Required) Specify a shell script or a command to run.",
            "displayType": "textarea",
            "default": "{{ssm:MyRunCommandParameter:2}}"
        }
    },
    "mainSteps": [
        {
            "action": "aws:runShellScript",
            "name": "RunScript",
            "inputs": {
                "runCommand": [
                    "{{commands}}"
                ]
            }
        }
    ]
}
```

Prise en charge des paramètres natifs pour Amazon Machine ImageIDs

Lorsque vous créez un paramètre String, vous pouvez spécifier le type de données comme `aws:ec2:image`, afin de vous assurer que la valeur du paramètre que vous saisissez est un format d'ID d'AMI (Amazon Machine Image) valide.

La prise en charge des formats d'ID d'AMI vous permet d'éviter de mettre à jour tous vos scripts et modèles avec un nouvel ID lors de chaque changement de l'AMI que vous souhaitez utiliser dans vos processus. Vous pouvez créer un paramètre avec le type de données `aws:ec2:image`, et saisir pour sa valeur l'ID d'une AMI. Il s'agit de l'AMI à partir de laquelle vous souhaitez actuellement créer de nouvelles instances. Vous référencez ensuite ce paramètre dans vos modèles, commandes et scripts.

Par exemple, vous pouvez spécifier le paramètre qui contient votre ID d'AMI préféré lorsque vous exécutez la commande Amazon EC2 `run-instances`.

> **Note**
>
> L'utilisateur qui exécute cette commande doit disposer des autorisations IAM qui incluent l'action d'API `ssm:GetParameters` pour que la valeur du paramètre soit validée. Sinon, le processus de création du paramètre échoue.

```
aws ec2 run-instances \
    --image-id resolve:ssm:/golden-ami \
    --count 1 \
    --instance-type t2.micro \
    --key-name my-key-pair \
```

```
    --security-groups my-security-group
```

Vous pouvez également choisir le Systems Manager qui contient votre AMI préférée lorsque vous créez une instance à l'aide de la console Amazon EC2. Pour plus d'informations, consultez Utilisation d'un Systems Manager paramètre pour trouver une AMI dans le Amazon EC2 Guide de l'utilisateur pour les instances Windows .

Lorsqu'il faut utiliser une AMI différente dans votre workflow de création d'instance, il vous suffit de mettre à jour le paramètre avec la nouvelle valeur d'AMI et Parameter Store valide à nouveau le fait que vous avez saisi l'ID dans le format approprié.

Fonctionnement de la validation du format d'AMI

Lorsque vous spécifiez `aws:ec2:image` comme type de données pour un paramètre, Systems Manager ne crée pas le paramètre immédiatement. Au lieu de cela, il effectue une opération de validation asynchrone pour s'assurer que la valeur du paramètre répond aux exigences de mise en forme d'un ID d'AMI, et que l'AMI spécifiée est disponible dans votre compte AWS.

Il est important de noter qu'un numéro de version de paramètre peut être généré avant la fin de l'opération de validation. Autrement dit, un numéro de version de paramètre généré seul n'indique pas que l'opération a été réalisée avec succès.

Pour savoir si vos paramètres sont créés correctement, nous vous recommandons d'utiliser Amazon EventBridge pour vous envoyer des notifications sur vos opérations de création et de mise à jour de paramètres. Ces notifications indiquent si une opération de paramètre a réussi ou non. Si une opération échoue, la notification inclut un message d'erreur indiquant la raison de l'échec.

```
{
    "version": "0",
    "id": "eed4a719-0fa4-6a49-80d8-8ac65EXAMPLE",
    "detail-type": "Parameter Store Change",
    "source": "aws.ssm",
    "account": "111122223333",
    "time": "2020-05-26T22:04:42Z",
    "region": "us-east-2",
    "resources": [
        "arn:aws:ssm:us-east-2:111122223333:parameter/golden-ami"
    ],
    "detail": {
        "exception": "Unable to Describe Resource",
        "dataType": "aws:ec2:image",
        "name": "golden-ami",
        "type": "String",
        "operation": "Create"
    }
}
```

Pour plus d'informations à propos de l'abonnement aux événementsParameter Store dans EventBridge, consultez Configuration de notifications ou d'actions de déclenchement basées sur des événements Parameter Store (p. 239).

Procédures Parameter Store

Les procédures décrites dans cette section vous montrent comment créer, stocker et exécuter des paramètres avec Parameter Store dans un environnement de test. Ces procédures pas à pas vous montrent comment utiliser Parameter Store avec d'autres fonctionnalités Systems Manager. Vous pouvez également utiliser Parameter Store avec d'autres services AWS. Pour plus d'informations, consultez la section Qu'est-ce qu'un paramètre ? (p. 220).

Table des matières

- Créer un SecureString paramètre et joindre une instance à un domaine (PowerShell) (p. 300)

Créer un SecureString paramètre et joindre une instance à un domaine (PowerShell)

Cette procédure pas à pas montre comment joindre une instance Windows Server à un domaine à l'aide de paramètres Systems Manager `SecureString` et Fonctionnalité Exécuter la commande. Elle a recours à des paramètres de domaine classiques, tels que le nom de domaine et un nom d'utilisateur de domaine. Ces valeurs sont transmises sous forme de valeurs de chaîne non chiffrées. Le mot de passe du domaine est chiffré à l'aide d'un clé CMK gérée par AWS et transmise en tant que chaîne chiffrée.

Prerequisites

Cette procédure suppose que vous avez déjà spécifié votre nom de domaine et l'adresse IP du serveur DNS dans le jeu d'options DHCP associé à votre Amazon VPC. Pour de plus amples informations, veuillez consulter Utilisation de jeux d'options DHCP dans le Amazon VPC Guide de l'utilisateur.

Pour créer un paramètre **`SecureString`** et attacher une instance à un domaine

1. Entrez les paramètres dans le système à l'aide des Outils AWS pour Windows PowerShell.

```
Write-SSMParameter -Name "domainName" -Value "DOMAIN-NAME" -Type String
Write-SSMParameter -Name "domainJoinUserName" -Value "DOMAIN\USERNAME" -Type String
Write-SSMParameter -Name "domainJoinPassword" -Value "PASSWORD" -Type SecureString
```

   > **Important**
   >
   > Seule la valeur d'un paramètre `SecureString` est chiffrée. Les noms de paramètre, les descriptions et les autres propriétés ne sont pas chiffrés.

2. Attachez les stratégies IAM suivantes aux autorisations de rôle IAM pour votre instance :

   - AmazonSSMManagedInstanceCore– Obligatoire. Cette stratégie gérée AWS permet à une instance gérée d'utiliser les fonctionnalités de base du service Systems Manager.
   - AmazonSSMDirectoryServiceAccess– Obligatoire. Cette stratégie gérée AWS permet à Agent SSM d'accéder à AWS Directory Service en votre nom pour les demandes de jonction de domaine par l'instance gérée.
   - Une stratégie personnalisée pour l'accès au compartiment S3 – Obligatoire. Agent SSM, qui se trouve sur votre instance et effectue des tâches Systems Manager, a besoin d'accéder à des compartiments S3 appartenant à Amazon spécifiques. Dans la stratégie de compartiment S3 personnalisée que vous créez, vous pouvez également fournir l'accès à vos propres compartiments S3 qui sont nécessaires pour les opérations Systems Manager.

     Exemples : Vous pouvez écrire la sortie des Fonctionnalité Exécuter la commande commandes ou Session Manager des sessions dans un compartiment S3, puis utiliser cette sortie ultérieurement à des fins d'audit ou de dépannage. Vous stockez des scripts d'accès ou des listes de références de correctifs personnalisées dans un compartiment S3, puis vous faites référence au script ou à la liste lorsque vous exécutez une commande, ou lorsqu'une référence de correctifs est appliquée.

     Pour plus d'informations sur la création d'une stratégie personnalisée pour l'accès à un compartiment Amazon S3, consultez Créer une stratégie de compartiment S3 personnalisée pour un profil d'instance (p. 36)

     > **Note**
     >
     > L'enregistrement des données de journal de sortie dans un compartiment S3 est facultatif, mais nous vous recommandons de le configurer au début de votre processus de

configuration de Systems Manager si vous avez décidé d'utiliser. Pour de plus amples informations, veuillez consulter Créer un compartiment dans le Amazon Simple Storage Service Guide de mise en route.

- CloudWatchAgentServerPolicy- Facultatif. Cette stratégie gérée AWS vous permet d'exécuter Agent CloudWatch sur des instances gérées. Cette stratégie permet de lire les informations sur une instance et de les écrire dans Amazon CloudWatch. Votre profil d'instance a besoin de cette stratégie uniquement si vous utilisez des services tels que EventBridge ou CloudWatch Logs .

  Note

  L'utilisation CloudWatch des EventBridge fonctionnalités et est facultative, mais nous vous recommandons de les configurer au début de votre processus de Systems Manager configuration si vous avez décidé de les utiliser. Pour plus d'informations, consultez le Guide de l'utilisateur Amazon EventBridge et le Amazon CloudWatch Logs User Guide.

3. Modifiez le rôle IAM attaché à l'instance et ajoutez la stratégie ci-dessous. Cette stratégie autorise l'instance à appeler l'API `kms:Decrypt`.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "kms:Decrypt"
         ],
         "Resource":[
            "arn:aws:kms:region:account-id:key/kms-key-id"
         ]
      }
   ]
}
```

4. Copiez et collez l'exemple json suivant dans un éditeur de texte simple et enregistrez le fichier comme `JoinInstanceToDomain.json` dans l'emplacement suivant : `c:\temp\JoinInstanceToDomain.json`.

```
{
    "schemaVersion": "2.2",
    "description": "Run a PowerShell script to securely domain-join a Windows Server
 instance",
    "mainSteps": [
        {
            "action": "aws:runPowerShellScript",
            "name": "runPowerShellWithSecureString",
            "precondition": {
                "StringEquals": [
                    "platformType",
                    "Windows"
                ]
            },
            "inputs": {
                "runCommand": [
                    "$domain = (Get-SSMParameterValue -Name
 domainName).Parameters[0].Value",
                    "if ((gwmi Win32_ComputerSystem).domain -eq $domain){write-host
 \"Computer is part of $domain, exiting\"; exit 0}",
                    "$username = (Get-SSMParameterValue -Name
 domainJoinUserName).Parameters[0].Value",
                    "$password = (Get-SSMParameterValue -Name domainJoinPassword -
WithDecryption $True).Parameters[0].Value | ConvertTo-SecureString -asPlainText -
Force",
```

```
                    "$credential = New-Object
 System.Management.Automation.PSCredential($username,$password)",
                    "Add-Computer -DomainName $domain -Credential $credential -
ErrorAction SilentlyContinue -ErrorVariable domainjoinerror",
                    "if($?){Write-Host \"Domain join succeeded, restarting\"; exit
 3010}else{Write-Host \"Failed to join domain with error:\" $domainjoinerror; exit 1 }"
                ]
            }
        }
    ]
}
```

5. Exécutez la commande suivante dans les Outils AWS pour Windows PowerShell pour créer un document SSM.

```
$json = Get-Content C:\temp\JoinInstanceToDomain | Out-String
New-SSMDocument -Name JoinInstanceToDomain -Content $json -DocumentType Command
```

6. Exécutez la commande suivante dans les Outils AWS pour Windows PowerShell pour joindre l'instance au domaine.

```
Send-SSMCommand -InstanceId instance-id -DocumentName JoinInstanceToDomain
```

Si la commande aboutit, le système renvoie des informations similaires à ce qui suit.

```
WARNING: The changes will take effect after you restart the computer EC2ABCD-EXAMPLE.
Domain join succeeded, restarting
Computer is part of example.local, exiting
```

Si la commande échoue, le système renvoie des informations similaires à ce qui suit.

```
Failed to join domain with error:
Computer 'EC2ABCD-EXAMPLE' failed to join domain 'example.local'
from its current workgroup 'WORKGROUP' with following error message:
The specified domain either does not exist or could not be contacted.
```

Audit et journalisation de l'activité de Parameter Store

AWS CloudTrailcapture les appels d'API effectués dans la AWS Systems Manager console , l' AWS Command Line Interface (AWS CLI) et le Systems Manager kit SDK . Vous pouvez voir les informations contenues dans la console CloudTrail ou dans un compartiment Amazon Simple Storage Service (Amazon S3). Un seul compartiment est utilisé pour tous les journaux CloudTrail de votre compte.

Les journaux des actions Parameter Store affichent les OpsItem activités de création, de mise à jour et de suppression. Pour de plus amples informations sur l'affichage et l'utilisation des journaux CloudTrail de l'activité Systems Manager, veuillez consulter Journalisation des appels d'API AWS Systems Manager avec AWS CloudTrail (p. 1336).

Pour de plus amples informations sur les options d'audit et de journalisation d'AWS Systems Manager, veuillez consulter Surveillance des AWS Systems Manager (p. 1326).

Dépannage de Parameter Store

Consultez les informations suivantes pour tenter de résoudre les problèmes liés à Parameter Store.

Dépannage pour la création de paramètres `aws:ec2:image`

Utilisez les informations suivantes pour résoudre les problèmes liés à la création de paramètres de type de données `aws:ec2:image`.

Rubriques

- EventBridge signale le message d'échec « Impossible de décrire la ressource » (p. 303)
- Le nouveau paramètre aws:ec2:image n'est pas disponible (p. 303)

EventBridge signale le message d'échec « Impossible de décrire la ressource »

Problème : Vous avez exécuté une commande pour créer un `aws:ec2:image` paramètre, mais la création du paramètre a échoué. Vous recevez une notification de EventBridge qui signale l'exception « Impossible de décrire la ressource ».

Solution: Ce message peut indiquer les éléments suivants :

- Vous n'avez pas reçu l'autorisation IAM pour l'action d'API `ec2:DescribeImages` ou vous n'avez pas l'autorisation d'accéder à l'image spécifique référencée dans le paramètre. Contactez un utilisateur IAM disposant des autorisations d'administrateur dans votre organisation pour demander les autorisations nécessaires.
- L'ID d'AMI que vous avez saisi en tant que valeur de paramètre n'est pas valide. Assurez-vous de saisir l'ID d'une AMI disponible dans la région AWS et dans le compte dans lesquels vous travaillez actuellement.

Le nouveau paramètre `aws:ec2:image` n'est pas disponible

Problème : Vous venez d'exécuter une commande pour créer un `aws:ec2:image` paramètre et un numéro de version a été signalé, mais le paramètre n'est pas disponible.

- Solution: Lorsque vous exécutez la commande pour créer un paramètre qui utilise le type `aws:ec2:image` de données, un numéro de version est immédiatement généré pour le paramètre, mais le format du paramètre doit être validé avant que le paramètre soit disponible. Ce processus peut prendre quelques minutes. Pour surveiller le processus de création et de validation des paramètres, vous pouvez faire ce qui suit :
  - Utilisez Amazon EventBridge pour vous envoyer des notifications sur vos opérations de création et de mise à jour des paramètres. Ces notifications indiquent si une opération de paramètre a réussi ou non. Pour plus d'informations à propos de l'abonnement aux événementsParameter Store dans EventBridge, consultez Configuration de notifications ou d'actions de déclenchement basées sur des événements Parameter Store (p. 239).
  - Dans la section Parameter Store de la console Systems Manager, mettez à jour périodiquement la liste des paramètres pour vérifier les détails des nouveaux paramètres ou de ceux mis à jour.
  - Utilisez la commande GetParameter pour vérifier le nouveau paramètre ou le paramètre mis à jour. Par exemple, en utilisant l'AWS CLI :

    ```
    aws ssm get-parameter --name MyParameter
    ```

    Pour un nouveau paramètre, un message `ParameterNotFound` est renvoyé jusqu'à ce que le paramètre soit validé. Pour un paramètre existant que vous mettez à jour, les informations sur la nouvelle version ne sont pas incluses tant que le paramètre n'est pas validé.

  Si vous tentez de créer ou de mettre à jour à nouveau le paramètre avant la fin du processus de validation, le système signale que la validation est toujours en cours. Si le paramètre n'est pas créé ou mis à jour avec succès, vous pouvez réessayer cinq minutes après la première tentative.

Actions et Change (Modifier) dans AWS Systems Manager

AWS Systems Manager fournit les fonctions suivantes pour exécuter des actions sur vos ressources AWS ou les modifier.

Rubriques

- Automatisation AWS Systems Manager (p. 304)
- AWS Systems Manager Modifier le calendrier (p. 680)
- AWS Systems Manager Fenêtres de maintenance (p. 688)

Automatisation AWS Systems Manager

Systems Manager Automation simplifie les tâches courantes de maintenance et de déploiement des instances EC2 et d'autres ressources AWS. Automation vous permet d'effectuer les tâches suivantes.

- Créer des flux de travail Automation pour configurer et gérer des instances et des ressources AWS.
- Créer des flux de travail personnalisés ou utiliser des flux de travail prédéfinis gérés par AWS.
- Recevoir des notifications concernant les tâches et les flux de travail Automation à l'aide d'Amazon EventBridge.
- Surveiller la progression d'Automation et les détails d'exécution via Amazon EC2 ou la console AWS Systems Manager.

Composants principaux

AWS Systems Manager Automation utilise les composants suivants pour exécuter des flux de travail d'automatisation.

Concept	Details (Détails)
Document Automation	Un document Systems Manager Automation définit le flux de travail Automation (actions exécutées par Systems Manager sur vos instances gérées et ressources AWS). L'outil Automation inclut plusieurs documents Automation prédéfinis que vous pouvez utiliser afin d'effectuer des tâches courantes, telles que le redémarrage d'une ou de plusieurs instances EC2 ou la création d'une Amazon Machine Image (AMI). Vous pouvez également créer vos propres documents Automation. Les documents utilisent JavaScript le format JSON (Object Notation) ou YAML et incluent les étapes et les paramètres que vous spécifiez. Les étapes sont exécutées par ordre séquentiel. Pour plus d'informations, consultez la section Utilisation des documents Automation (p. 446). Les documents Automation sont des documents Systems Manager de type `Automation`,

Concept	Details (Détails)
	contrairement aux documents `Command`, `Policy` et `Session`. Les documents Automation prennent en charge actuellement la version de schéma 3.0. Les documents de commande utilisent la version de schéma 1.2, 2.0, ou 2.2. Les documents de stratégie utilisent la version de schéma 2.0 ou ultérieure.
Action Automation	Le flux de travail Automation défini dans un document d'Automation comprend une ou plusieurs étapes. Chaque étape est associée à une action spécifique. L'action détermine les entrées, le comportement et les sorties de l'étape. Les étapes sont définies dans la section `mainSteps` de votre document d'Automation. Automation prend en charge 20 types d'action différents. Pour de plus amples informations, veuillez consulter Référence sur les actions Systems Manager Automation (p. 386).
Quota d'automatisation	Chaque AWS compte peut exécuter 100 automatisations simultanément. Cela inclut les automatisations enfants (automatisations qui sont lancées par une autre automatisation) et les automatisations de contrôle du débit. Si vous essayez d'exécuter plus d'automatisations que cela, Systems Manager ajoute les automatisations supplémentaires à une file d'attente et affiche le statut En attente. Pour plus d'informations sur l'exécution d'automatisations, consultez Exécution d'une automatisation simple (p. 314) .
Quota de file d'attente d'Automation	Si vous essayez d'exécuter plus d'automatisations que la limite d'automatisations simultanées, les automatisations suivantes sont ajoutées à une file d'attente. Chaque AWS compte peut mettre en file d'attente 1 000 automatisations. Lorsqu'une automatisation se termine (ou atteint un état terminal), la première automatisation dans la file d'attente est démarrée.
Quota d'automatisation du contrôle du débit	Chaque AWS compte peut exécuter simultanément 25 automatisations de contrôle de débit. Si vous essayez d'exécuter plus d'automatisations de contrôle de débit que la limite d'automatisation de contrôle de débit simultanée, Systems Manager ajoute les automatisations de contrôle de débit suivantes à une file d'attente et affiche le statut En attente. Pour plus d'informations sur l'exécution d'automatisations de contrôle de débit, consultez Exécution d'automatisations qui utilisent des cibles et des contrôles du débit (p. 328) .

Concept	Details (Détails)
Quota de file d'attente d'automatisation de contrôle du débit	Si vous essayez d'exécuter plus d'automatisations que la limite d'automatisation du contrôle de débit simultané, les automatisations suivantes sont ajoutées à une file d'attente. Chaque AWS compte peut mettre en file d'attente 1 000 automatisations de contrôle du débit. Lorsqu'une automatisation se termine (ou atteint un état terminal), la première automatisation dans la file d'attente est démarrée.

Cas d'utilisation Automation

Cette section inclut des cas d'utilisation courants pour AWS Systems Manager Automation.

Exécutez des tâches informatiques courantes

Automation peut simplifier des tâches informatiques courantes, telles que la modification de l'état d'une ou plusieurs instances (à l'aide d'un flux de travail d'approbation) et la gestion des états de l'instance en fonction d'un planning. Voici quelques exemples :

- Utilisez le document AWS-StopEC2InstanceWithApproval pour demander qu'un ou plusieurs utilisateurs AWS Identity and Access Management (IAM) approuvent l'action d'arrêt de l'instance. Après la réception de l'approbation, Automation arrête l'instance.
- Utilisez le document AWS-StopEC2Instance pour arrêter automatiquement des instances selon un calendrier spécifique à l'aide d'Amazon EventBridge ou d'une tâche de fenêtre de maintenance. Par exemple, vous pouvez configurer un flux de travail d'automatisation pour arrêter des instances tous les vendredis soir, puis les redémarrer tous les lundis matin.
- Utilisez le document AWS-UpdateCloudFormationStackWithApproval pour mettre à jour les ressources qui ont été déployées à l'aide CloudFormation du modèle . La mise à jour applique un nouveau modèle. Vous pouvez configurer Automation pour demander l'approbation par un ou plusieurs utilisateurs IAM avant le début de la mise à jour.

Pour de plus amples informations sur la façon d'exécuter un flux de travail Automation à l'aide de State Manager, veuillez consulter Exécution d'automatisations avec des déclencheurs à l'aide deState Manager (p. 349).

Exécutez simultanément des tâches perturbatrices en toute sécurité

Systems Manager inclut des fonctions qui vous aident à cibler de grands groupes d'instances à l'aide de balises Amazon EC2, et des contrôles de rapidité qui vous aident à déployer des modifications selon les limites que vous définissez.

Utilisez le document AWS-RestartEC2InstanceWithApproval pour cibler un groupe de ressources AWS incluant plusieurs instances. Vous pouvez configurer le flux de travail Automation pour qu'il utilise des contrôles de rapidité. Par exemple, vous pouvez spécifier le nombre d'instances qui doivent être redémarrées simultanément. Vous pouvez également spécifier un nombre maximal d'erreurs autorisées avant l'annulation du flux de travail Automation.

Simplifiez des tâches complexes

Automation propose des automatisations en un clic des tâches complexes, telles que la création d'Amazon Machines Images (AMI) finales et la récupération d'instances EC2 inaccessibles. Voici quelques exemples :

- Utilisez les `AWS-UpdateLinuxAmi` documents `AWS-UpdateWindowsAmi` et pour créer un Golden AMIs à partir d'une AMI source. Vous pouvez exécuter des scripts personnalisés avant et après

l'application des mises à jour. Vous pouvez également inclure ou exclure l'installation de packages spécifiques. Pour découvrir comment exécuter ces flux de travail grâce à divers exemples, veuillez consulter Procédures d'automatisation (p. 632).

- Utilisez le document AWSSupport -ExecuteEC2Rescue pour récupérer les instances dégradées. Une instance peut devenir inaccessible pour différentes raisons, notamment des erreurs de configuration réseau, des problèmes RDP ou des paramètres de pare-feu. Le dépannage et la récupération de l'accès à l'instance nécessitait auparavant des dizaines d'étapes manuelles avant le rétablissement de l'accès. Le document AWSSupport -ExecuteEC2Rescue vous permet de récupérer l'accès en spécifiant un ID d'instance et en cliquant sur un bouton. Pour découvrir comment exécuter ce flux de travail sur la base d'un exemple, veuillez consulter Procédure : Exécuter l'EC2Rescueoutil sur des instances inaccessibles (p. 656).

Améliorez la sécurité opérationnelle

L'administration déléguée permet de limiter ou d'élever les autorisations utilisateur pour divers types de tâches.

Elle vous donne la possibilité de fournir des autorisations pour certaines tâches sur une ressource donnée, sans avoir à accorder directement à un utilisateur une autorisation d'accès aux ressources. Vous améliorez ainsi votre profil de sécurité global. Par exemple, supposons que l'Utilisateur 1 n'ait pas le droit de redémarrer des instances EC2, mais que vous souhaitiez lui accorder l'autorisation de le faire. Au lieu d'accorder des autorisations directes à l'Utilisateur 1, vous pouvez :

- Créer un rôle IAM avec les autorisations requises pour arrêter et démarrer des instances EC2.
- Créer un document d'Automation et y intégrer le rôle. (Pour ce faire, la méthode la plus simple consiste à personnaliser le document AWS-RestartEC2Instance et à y intégrer le rôle au lieu d'affecter un rôle de service d'automatisation [ou le rôle de responsable]).
- Modifiez les autorisations IAM pour l'Utilisateur 1 et accordez à l'utilisateur l'autorisation d'exécuter le document.

Pour découvrir comment déléguer l'accès à un flux de travail Automation grâce à un exemple, veuillez consulter Exécution d'une automatisation à l'aide de l'administration déléguée (p. 372).

Partagez les bonnes pratiques

Automation vous permet de partager les bonnes pratiques avec le reste de votre organisation.

Vous pouvez créer des bonnes pratiques pour la gestion des ressources dans les documents Automation et partager facilement les documents entre les régions et les groupes AWS. Vous pouvez également limiter les valeurs autorisées pour les paramètres acceptés par le document.

Prise en charge de Amazon EventBridge

Cette fonction Systems Manager est prise en charge en tant que type de cible dans les règles EventBridge. Pour plus d'informations, consultez Surveillance des événements Systems Manager avec Amazon EventBridge (p. 1340) et Référence : modèles et types Amazon EventBridge d'événements pourSystems Manager (p. 1387).

Table des matières

- Configuration d'Automation (p. 308)
- Utilisation des automatisations (p. 313)
- Référence sur les actions Systems Manager Automation (p. 386)
- Utilisation des documents Automation (p. 446)
- Référence des documents Systems Manager Automation (p. 511)
- Procédures d'automatisation (p. 632)
- Dépannage de Systems Manager Automation (p. 676)

Configuration d'Automation

Pour configurer Automation, vous devez vérifier l'accès utilisateur au service Automation et configurer les rôles de manière à ce que le service puisse effectuer des actions sur vos ressources. Pour garantir un accès correct à Systems Manager Automation, passez en revue les exigences suivantes relatives au rôle d'utilisateur et de service.

Vérification de l'accès des utilisateurs aux flux de travail Automation

Vérifiez que vous avez l'autorisation d'exécuter des flux de travail Automation. Si votre compte utilisateur, votre groupe ou votre rôle AWS Identity and Access Management (IAM) dispose des autorisations d'administrateur, vous avez accès à Systems Manager Automation. Si vous ne disposez pas des autorisations d'administrateur, un administrateur doit vous les donner en affectant la stratégie gérée `AmazonSSMFullAccess` ou une stratégie dotée d'autorisations de même type à votre compte, groupe ou rôle IAM.

> **Important**
>
> La stratégie IAM `AmazonSSMFullAccess` accorde des autorisations pour les actions Systems Manager. Toutefois, certains documents Automation nécessitent des autorisations pour d'autres services, tels que le document `AWS-ReleaseElasticIP` , qui nécessite IAM des autorisations pour `ec2:ReleaseAddress` . Par conséquent, vous devez passer en revue les actions réalisées dans un document Automation pour vous assurer que votre compte IAM utilisateur, votre groupe ou votre rôle dispose des autorisations nécessaires pour effectuer les actions incluses dans le document.

Configuration d'un accès à un rôle de service (rôle de responsable) pour les flux de travail Automation

Les flux de travail Automation peuvent être lancés dans le contexte d'un rôle de service (ou rôle de responsable). Cela permet au service d'effectuer des actions en votre nom. Si vous ne spécifiez pas de rôle de responsable, Automation utilise le contexte de l'utilisateur qui a appelé l'exécution.

Cependant, les situations suivantes nécessitent que vous spécifiiez un rôle du service pour Automation :

- Lorsque vous souhaitez restreindre les privilèges d'un utilisateur sur une ressource tout en souhaitant que l'utilisateur puisse exécuter un flux de travail d'automatisation nécessitant des privilèges supérieurs. Dans ce scénario, vous pouvez créer un rôle de service avec des privilèges supérieurs et autoriser l'utilisateur à exécuter le flux de travail.
- Lorsque vous créez une association State Manager qui exécute un flux de travail Automation.
- Lorsque vous prévoyez que la durée d'exécution de certaines opérations dépasse 12 heures.
- Lorsque vous exécutez un document Automation qui n'appartient pas à Amazon et qui utilise l'action `aws:executeScript` pour appeler une opération d'API AWS ou pour agir sur une ressource AWS. Pour plus d'informations, consultez la section Autorisations pour exécuter des exécutions Automation (p. 451).

Si vous avez besoin de créer un rôle de service pour Automation, vous pouvez utiliser l'une des méthodes suivantes.

Rubriques

- Méthode 1 : Utiliser AWS CloudFormation pour configurer un rôle de service pour Automation (p. 309)
- Méthode 2 : Utiliser IAM pour configurer des rôles pour Automation (p. 310)

Méthode 1 : Utiliser AWS CloudFormation pour configurer un rôle de service pour Automation

Vous pouvez créer un rôle de service pour Automation à partir d'un modèle AWS CloudFormation. Après avoir créé le rôle de service, vous pouvez spécifier le rôle de service dans les flux de travail Automation à l'aide du paramètre `AutomationAssumeRole` . Pour plus d'informations sur l'exécution d'un flux de travail Automation à l'aide du rôle de service Automation, consultez Exécution d'une automatisation à l'aide d'un rôle IAM de service (p. 368) .

Rubriques

- Création du rôle de service via AWS CloudFormation (p. 309)
- Copier les informations de rôle pour Automation (p. 310)

Création du rôle de service via AWS CloudFormation

Utilisez la procédure suivante pour créer le rôle IAM requis pour Systems Manager Automation via AWS CloudFormation.

Pour créer le rôle IAM requis

1. Téléchargez le dossier AWS-SystemsManager-AutomationServiceRole.zip. Ce dossier inclut le fichier modèle AWS CloudFormation `AWS-SystemsManager-AutomationServiceRole.yaml`.
2. Ouvrez la console AWS CloudFormation à l'adresse https://console.aws.amazon.com/cloudformation.
3. Choisissez Créer une pile.
4. Dans la section Spécifier un modèle, choisissez Charger un modèle de fichier.
5. Choisissez Browse (Parcourir), puis choisissez le fichier modèle AWS CloudFormation `AWS-SystemsManager-AutomationServiceRole.yaml`.
6. Choisissez Nex t(Suivant).
7. Dans la section Spécifier les détails, entrez un nom dans le champ Nom de la pile.
8. Sur la page Configurer les options de pile, vous n'avez pas besoin d'effectuer de sélections. Choisissez Nex t(Suivant).
9. Faites défiler la page Vérification vers le bas et choisissez l'option Je comprends qu'AWS CloudFormation peut créer des ressources IAM.
10. Sélectionnez Create.

AWS CloudFormation affiche le statut CREATE_IN_PROGRESS pendant environ trois minutes. Le statut devient CREATE_COMPLETE une fois que la pile a été créée et que vos rôles sont prêts à être utilisés.

> **Important**
>
> Si vous exécutez un flux de travail d'automatisation qui appelle d'autres services à l'aide d'un rôle de service AWS Identity and Access Management (IAM), le rôle de service doit être configuré avec l'autorisation d'appeler ces services. Cette exigence s'applique à tous les documents AWS Automation (documents `AWS-*`) comme les documents `AWS-ConfigureS3BucketLogging`, `AWS-CreateDynamoDBBackup` et `AWS-RestartEC2Instance`, pour n'en citer que quelques-uns. Cette exigence s'applique également à tous les documents Automation personnalisés que vous créez qui appellent d'autres services AWS à l'aide d'actions qui appellent d'autres services. Par exemple, si vous utilisez les actions `aws:executeAwsApi`, `aws:createStack` ou `aws:copyImage`, vous devez configurer le rôle de service avec l'autorisation d'appeler ces services. Vous pouvez activer des autorisations à d'autres services AWS en ajoutant une stratégie IAM en ligne au rôle. Pour de plus amples informations, veuillez consulter (Facultatif) Ajoutez une stratégie en ligne Automation pour appeler d'autres services AWS (p. 311).

Copier les informations de rôle pour Automation

Utilisez la procédure suivante pour copier des informations relatives au rôle de service Automation à partir de la console AWS CloudFormation. Vous devez spécifier ces rôles lorsque vous exécutez un document d'Automation.

> **Note**
>
> Vous n'avez pas besoin de copier les informations du rôle en utilisant cette procédure si vous exécutez les documents `AWS-UpdateLinuxAmi` ou `AWS-UpdateWindowsAmi`. Ces documents possèdent déjà les rôles requis spécifiés comme valeurs par défaut. Ces rôles spécifiés dans ces documents utilisant des stratégies gérées IAM.

Pour copier les noms de rôle

1. Ouvrez la console AWS CloudFormation à l'adresse https://console.aws.amazon.com/cloudformation.
2. Sélectionnez le Nom de la pile d'automatisation que vous avez créé lors de la procédure précédente.
3. Choisissez l'onglet Ressources.
4. Choisissez le lien ID physique pour AutomationServiceRole . La IAM console s'ouvre sur un résumé du rôle de service Automation.
5. Copiez l'ARN (Amazon Resource Name) en regard de l'ARN de rôle. L'ARN est similaire à ce qui suit : `arn:aws:iam::12345678:role/AutomationServiceRole`
6. Collez l'ARN dans un fichier texte à utiliser ultérieurement.

La configuration du rôle de service pour Automation est terminée. Vous pouvez désormais utiliser l'ARN du rôle de service Automation dans vos documents Automation.

Méthode 2 : Utiliser IAM pour configurer des rôles pour Automation

Si vous avez besoin de créer un rôle de service pour Systems Manager Automation, exécutez les tâches suivantes. Pour plus d'informations sur le moment où un rôle de service est requis pour Automation, consultez Configuration d'Automation (p. 308).

Tâches

- Tâche 1 : Créer un rôle de service pour Automation (p. 310)
- Tâche 2 : Attachement de la stratégie iam:PassRole à votre rôle Automation (p. 312)
- Tâche 3 : Configuration de l'accès utilisateur à Automation (p. 313)

Tâche 1 : Créer un rôle de service pour Automation

Utilisez la procédure suivante pour créer un rôle de service (ou un rôle de responsable) pour Systems Manager Automation.

> **Note**
>
> Vous pouvez également utiliser ce rôle dans les documents Automation, par exemple, le document `AWS-CreateManagedLinuxInstance`. L'utilisation de ce rôle ou de l' ARNs dans les documents Automation permet à Automation d'effectuer des actions dans votre environnement, telles que lancer de nouvelles instances et effectuer des actions en votre nom.

Pour créer un rôle IAM et autoriser Automation à l'assumer

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.

2. Dans le volet de navigation, choisissez Rôles, puis Créer un rôle.
3. Sous Select type of trusted entity (Sélectionner le type d'entité de confiance), choisissez Service AWS.
4. Dans la section Choose a use case (Choisir un cas d'utilisation), choisissez Systems Manager, puis Next (Suivant) : Autorisations.
5. Sur la page Attached permissions policy (Stratégie d'autorisations attachée), recherchez la AmazonSSMAutomationRole stratégie, choisissez-la, puis choisissez Next (Suivant) : Passez en revue.
6. Sur la page Vérification, saisissez un nom dans la zone Nom du rôle, puis saisissez une description.
7. Sélectionnez Créer un rôle. Le système vous renvoie à la page Rôles.
8. Sur la page Rôles, choisissez le rôle que vous venez de créer pour ouvrir la page Récapitulatif. Notez le Nom du rôle et l'ARN de rôle. Vous spécifierez l'ARN du rôle lorsque vous attacherez la stratégie iam:PassRole à votre compte IAM dans la procédure suivante. Vous pouvez également spécifier le nom du rôle et l'ARN dans les documents Automation.

   Note

   La AmazonSSMAutomationRole stratégie attribue l'autorisation du rôle Automation à un sous-ensemble de AWS Lambda fonctions au sein de votre compte. Ces fonctions commencent par « Automation ». Si vous avez l'intention d'utiliser Automation avec des fonctions Lambda, l'ARN Lambda doit utiliser le format suivant :
   `"arn:aws:lambda:*:*:function:Automation*"`
   Si vous avez Lambda des fonctions existantes dont vous n'utilisez pas ce format, vous devez également attacher une ARNs stratégie supplémentaire à votre rôle d'automatisation, telle que la Lambda AWSLambdaRole stratégie. La stratégie ou le rôle supplémentaire doit fournir un accès plus large aux fonctions Lambda au sein du compte AWS.

(Facultatif) Ajoutez une stratégie en ligne Automation pour appeler d'autres services AWS

Si vous exécutez une instance d'Automation qui appelle d'autres services AWS à l'aide d'un rôle de service IAM, le rôle de service doit être configuré avec l'autorisation d'appeler ces services. Cette exigence s'applique à tous les documents AWS Automation (documents AWS-*) tels que `AWS-ConfigureS3BucketLogging`, `AWS-CreateDynamoDBBackup` et `AWS-RestartEC2Instance`, pour n'en citer que quelques-uns. Cette exigence s'applique également à tous les documents Automation personnalisés que vous créez qui appellent d'autres services AWS à l'aide d'actions qui appellent d'autres services. Par exemple, si vous utilisez les actions `aws:executeAwsApi`, `aws:CreateStack` ou `aws:copyImage`, pour n'en citer que quelques-unes, vous devez configurer le rôle de service avec l'autorisation d'appeler ces services. Vous pouvez activer des autorisations à d'autres services AWS en ajoutant une stratégie IAM en ligne au rôle.

Pour intégrer une stratégie en ligne à un rôle de service (console IAM)

1. Connectez-vous à la AWS Management Console et ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le volet de navigation, choisissez Roles (Rôles).
3. Dans la liste, choisissez le nom du rôle que vous souhaitez modifier.
4. Choisissez l'onglet Autorisations.
5. Choisissez Add inline policy (Ajouter une stratégie en ligne).
6. Choisissez l'onglet JSON.
7. Entrez un document de stratégie JSON pour les services AWS que vous voulez appeler. Voici deux exemples de document de stratégie JSON.

   Amazon S3 PutObject et GetObject exemple

```
{
```

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

Amazon EC2 CreateSnapshot et DescribeSnapShots exemple

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":"ec2:CreateSnapshot",
         "Resource":"*"
      },
      {
         "Effect":"Allow",
         "Action":"ec2:DescribeSnapshots",
         "Resource":"*"
      }
   ]
}
```

Pour de plus amples informations sur le langage des stratégies IAM, veuillez consulter Référence de stratégie JSON IAM dans le IAM Guide de l'utilisateur.

8. Lorsque vous avez terminé, choisissez Examiner une stratégie. Le programme de validation des stratégies signale les éventuelles erreurs de syntaxe.
9. Sur la page Examiner une stratégie, entrez un Nom pour la stratégie que vous êtes en train de créer. Vérifiez le Summary (Récapitulatif) de la stratégie pour voir les autorisations accordées par votre stratégie. Choisissez ensuite Create policy (Créer une stratégie) pour enregistrer votre travail.
10. Une fois que vous avez créé une stratégie en ligne, elle est automatiquement intégrée à votre rôle.

Tâche 2 : Attachement de la stratégie iam:PassRole à votre rôle Automation

Utilisez la procédure suivante pour attacher la stratégie `iam:PassRole` à votre rôle de service Automation. Cela permet au service Automation de transmettre le rôle à d'autres services ou fonctionnalités Systems Manager lors de l'exécution de flux de travail d'automatisation.

Pour attacher la stratégie iam:PassRole à votre rôle d'automatisation

1. Dans la page Récapitulatif du rôle que vous venez de créer, choisissez l'onglet Autorisations.
2. Choisissez Add inline policy (Ajouter une stratégie en ligne).
3. Dans la page Créer une stratégie, choisissez l'onglet Éditeur visuel.
4. Choisissez Service, puis choisissez IAM.
5. Choisissez Sélectionner des actions.
6. Dans la zone de texte Filtrer les actions, tapez **`PassRole`** , puis choisissez l'PassRoleoption.
7. Choisissez Ressources. Vérifiez que Spécifique est sélectionné, puis choisissez Add ARN (Ajouter l'ARN).

8. Dans le champ Specify ARN for role (Spécifier l'ARN du rôle), collez l'ARN du rôle Automation que vous avez copié à la fin de la tâche 1. Le système remplit automatiquement les champs Compte et Role name with path (Nom du rôle avec chemin d'accès).

   Note

   Si vous souhaitez que le rôle de service Automation attache un rôle de profil d'instance IAM à une instance EC2, vous devez ajouter l'ARN du rôle de profil d'instance IAM. Cela permet au rôle de service Automation de transmettre le rôle de profil d'instance IAM à l'instance EC2 cible.
9. Choisissez Add (Ajouter).
10. Choisissez Examiner une stratégie.
11. Sur la page Review Policy (Examiner une stratégie), tapez un nom, puis choisissez Create Policy (Créer une stratégie).

Tâche 3 : Configuration de l'accès utilisateur à Automation

Si votre compte utilisateur, votre groupe ou votre rôle AWS Identity and Access Management (IAM) dispose des autorisations d'administrateur, vous avez accès à Systems Manager Automation. Si vous ne disposez pas des autorisations d'administrateur, un administrateur doit vous les donner en affectant la stratégie gérée `AmazonSSMFullAccess` ou une stratégie dotée d'autorisations de même type à votre compte, groupe ou rôle IAM.

Utilisez la procédure suivante afin de configurer un compte utilisateur pour utiliser Automation. Le compte utilisateur que vous choisissez aura l'autorisation de configurer et d'exécuter Automation. Si vous devez créer un compte utilisateur, veuillez consulter la section Création d'un utilisateur IAM dans votre compte AWS dans le IAM Guide de l'utilisateur.

Pour configurer l'accès utilisateur et associer la stratégie iam:PassRole à un compte utilisateur

1. Dans le panneau de navigation IAM, choisissez Utilisateurs, puis choisissez le compte utilisateur que vous voulez configurer.
2. Sous l'onglet Autorisations, dans la liste des stratégies, vérifiez que la stratégie **`AmazonSSMFullAccess`** est répertoriée ou qu'une stratégie comparable autorise le compte à accéder à Systems Manager.
3. Choisissez Add inline policy (Ajouter une stratégie en ligne).
4. Sur la page Create policy (Créer une stratégie) choisissez Visual Editor (Éditeur visuel), puis Choose a service (Choisir un service).
5. Pour Services AWS, choisissez AWS Identity and Access Management.
6. Pour Actions, entrez PassRole dans l'invite Filtrer les actions, puis choisissez PassRole .
7. Dans la section Resources (Ressources) choisissez Add ARN (Ajouter un ARN), collez l'ARN du rôle de service Automation que vous avez copié à la fin de la tâche 1, puis choisissez Add (Ajouter).
8. Choisissez Examiner une stratégie.
9. Sur la page Review Policy (Examiner une stratégie) indiquez un nom pour la stratégie, puis choisissez Create policy (Créer une stratégie).

Vous avez fini de configurer les rôles requis pour Automation. Vous pouvez désormais utiliser l'ARN du rôle de service Automation dans vos documents Automation.

Utilisation des automatisations

Cette section comprend des informations sur le mode d'exécution des flux de travail Systems Manager Automation. Pour plus d'exemples sur la façon d'exécuter des automatisations, consultez Procédures d'automatisation (p. 632) .

Table des matières

- Exécution d'une automatisation simple (p. 314)
- Exécution manuelle d'une automatisation (p. 318)
- Exécution d'une automatisation avec des approbateurs (p. 324)
- Exécution d'automatisations qui utilisent des cibles et des contrôles du débit (p. 328)
- Exécution d'automatisations basées sur des déclencheurs (p. 343)
- Exécution d'automatisations à l'aide de différents modèles de sécurité (p. 364)
- Exécution d'automatisations dans plusieurs AWS comptes et régions (p. 376)

Exécution d'une automatisation simple

Les procédures suivantes expliquent comment exécuter une Systems Manager automatisation simple à l'aide de la AWS Systems Manager console , AWS Command Line Interface de l' (AWS CLI) et Outils AWS pour Windows PowerShell d' . Le flux de travail s'exécute dans le contexte de l'utilisateur AWS Identity and Access Management (IAM) actif. Cela signifie que vous n'avez pas besoin de configurer IAM des autorisations supplémentaires tant que vous avez le droit d'exécuter le document Automation et les actions appelées par le document. Si vous disposez d'autorisations d'administrateur dans IAM , vous avez déjà l'autorisation d'exécuter cette automatisation.

> **Note**
>
> Pour plus d'informations sur l'exécution d'une automatisation qui utilise un rôle de IAM service ou des formes plus avancées d'administration déléguée, consultez Exécution d'automatisations à l'aide de différents modèles de sécurité (p. 364) .

Exécution d'une automatisation simple (console)

La procédure suivante décrit comment utiliser la Systems Manager console pour exécuter une automatisation simple.

Pour exécuter une automatisation simple

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Automation (Automatisation), puis Execute automation (Exécuter l'automatisation).
3. Dans la liste Automation document (Document d'automatisation), choisissez un document. Choisissez une ou plusieurs options dans le volet Document categories (Catégories de documents) pour filtrer les documents SSM en fonction de leur objectif. Pour afficher un document dont vous êtes propriétaire, sélectionnez l’onglet Owned by me (M’appartenant). Pour afficher un document partagé avec votre compte, sélectionnez l’onglet Shared with me (Partagé avec moi). Pour afficher tous les documents, sélectionnez l’onglet All documents (Tous les documents).

   > **Note**
   >
   > Pour afficher des informations sur un document, sélectionnez son nom.

4. Dans la section Document details (Détails du document), vérifiez que l'option Document version (Version de document) correspond à la version que vous souhaitez exécuter. Le système inclut les options de version suivantes :

   - Default version at runtime (Version par défaut lors de l'exécution) : choisissez cette option si le document d'automatisation est mis à jour régulièrement et qu'une nouvelle version par défaut est affectée.

- Latest version at runtime (Dernière version lors de l'exécution) : choisissez cette option si le document Automation est mis à jour régulièrement et que vous souhaitez exécuter la version la plus récemment mise à jour.
- 1 (Default) [1 (Par défaut)] : choisissez cette option pour exécuter la première version du document, qui est la version par défaut.

5. Choisissez Suivant.
6. Dans la section Execution Mode (Mode d'exécution), choisissez Simple execution (Exécution simple).
7. Dans la section Input parameters (Paramètres d'entrée), spécifiez les entrées obligatoires. Le cas échéant, vous pouvez choisir un rôle de service IAM dans la liste AutomationAssumeRole.
8. Sélectionnez Exécuter.

La console affiche le statut de l'automatisation. Si l'exécution d'Automation échoue, veuillez consulter Dépannage de Systems Manager Automation (p. 676).

Exécution d'une automatisation simple (ligne de commande)

La procédure suivante décrit comment utiliser l' AWS CLI (sous Linux ou Windows) ou Outils AWS pour PowerShell pour exécuter une automatisation simple.

Pour exécuter une automatisation simple

1. Installez et configurez l'AWS CLI ou Outils AWS pour PowerShell si vous ne l'avez pas déjà fait.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).
2. Exécutez la commande suivante pour démarrer une automatisation simple.

   Linux

   ```
   aws ssm start-automation-execution \
       --document-name DocumentName \
       --parameters ParametersRequiredByDocument
   ```

   Windows

   ```
   aws ssm start-automation-execution ^
       --document-name DocumentName ^
       --parameters ParametersRequiredByDocument
   ```

   PowerShell

   ```
   Start-SSMAutomationExecution `
     -DocumentName DocumentName `
     -Parameter ParametersRequiredByDocument
   ```

   Voici un exemple d'utilisation du document `AWS-RestartEC2Instance` pour redémarrer l'instance EC2 spécifiée.

   Linux

   ```
   aws ssm start-automation-execution \
       --document-name "AWS-RestartEC2Instance" \
       --parameters "InstanceId=i-1234567890abcdef0"
   ```

Windows

```
aws ssm start-automation-execution ^
    --document-name "AWS-RestartEC2Instance" ^
    --parameters "InstanceId=i-1234567890abcdef0"
```

PowerShell

```
Start-SSMAutomationExecution `
  -DocumentName AWS-RestartEC2Instance `
  -Parameter @{"InstanceId"="i-1234567890abcdef0"}
```

Le système renvoie des informations similaires à celles qui suivent.

Linux

```
{
    "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0123456789ab"
}
```

Windows

```
{
    "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0123456789ab"
}
```

PowerShell

```
4105a4fc-f944-11e6-9d32-0123456789ab
```

3. Exécutez la commande suivante pour récupérer le statut de l'automatisation.

Linux

```
aws ssm describe-automation-executions \
    --filter "Key=ExecutionId,Values=4105a4fc-f944-11e6-9d32-0123456789ab"
```

Windows

```
aws ssm describe-automation-executions ^
    --filter "Key=ExecutionId,Values=4105a4fc-f944-11e6-9d32-0123456789ab"
```

PowerShell

```
Get-SSMAutomationExecutionList | `
  Where {$_.AutomationExecutionId -eq "4105a4fc-f944-11e6-9d32-0123456789ab"}
```

Le système renvoie des informations similaires à celles qui suivent.

Linux

```
{
    "AutomationExecutionMetadataList": [
```

```
        {
            "AutomationExecutionStatus": "InProgress",
            "CurrentStepName": "stopInstances",
            "Outputs": {},
            "DocumentName": "AWS-RestartEC2Instance",
            "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0123456789ab",
            "DocumentVersion": "1",
            "ResolvedTargets": {
                "ParameterValues": [],
                "Truncated": false
            },
            "AutomationType": "Local",
            "Mode": "Auto",
            "ExecutionStartTime": 1564600648.159,
            "CurrentAction": "aws:changeInstanceState",
            "ExecutedBy": "arn:aws:sts::123456789012:assumed-role/Administrator/
Admin",
            "LogFile": "",
            "Targets": []
        }
    ]
}
```

Windows

```
{
    "AutomationExecutionMetadataList": [
        {
            "AutomationExecutionStatus": "InProgress",
            "CurrentStepName": "stopInstances",
            "Outputs": {},
            "DocumentName": "AWS-RestartEC2Instance",
            "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0123456789ab",
            "DocumentVersion": "1",
            "ResolvedTargets": {
                "ParameterValues": [],
                "Truncated": false
            },
            "AutomationType": "Local",
            "Mode": "Auto",
            "ExecutionStartTime": 1564600648.159,
            "CurrentAction": "aws:changeInstanceState",
            "ExecutedBy": "arn:aws:sts::123456789012:assumed-role/Administrator/
Admin",
            "LogFile": "",
            "Targets": []
        }
    ]
}
```

PowerShell

```
AutomationExecutionId       : 4105a4fc-f944-11e6-9d32-0123456789ab
AutomationExecutionStatus   : InProgress
AutomationType              : Local
CurrentAction               : aws:changeInstanceState
CurrentStepName             : startInstances
DocumentName                : AWS-RestartEC2Instance
DocumentVersion             : 1
ExecutedBy                  : arn:aws:sts::123456789012:assumed-role/Administrator/
Admin
ExecutionEndTime            : 1/1/0001 12:00:00 AM
ExecutionStartTime          : 7/31/2019 7:17:28 PM
```

```
FailureMessage              :
LogFile                     :
MaxConcurrency              :
MaxErrors                   :
Mode                        : Auto
Outputs                     : {}
ParentAutomationExecutionId :
ResolvedTargets             : Amazon.SimpleSystemsManagement.Model.ResolvedTargets
Target                      :
TargetMaps                  : {}
TargetParameterName         :
Targets                     : {}
```

Exécution manuelle d'une automatisation

Les procédures suivantes expliquent comment utiliser la console AWS Systems Manager, l'AWS Command Line Interface (AWS CLI) et les Outils AWS pour Windows PowerShell pour exécuter un flux de travail Systems Manager Automation à l'aide du mode d'exécution manuelle. En utilisant le mode d'exécution manuelle, l'automatisation démarre à l'état Waiting et s'interrompt à l'état Waiting entre chaque étape. Ceci vous permet de contrôler quand le flux de travail se poursuit, ce qui est utile si vous avez besoin de lire le résultat d'une étape avant de continuer.

Le flux de travail s'exécute dans le contexte de l'utilisateur AWS Identity and Access Management (IAM) actif. Cela signifie que vous n'avez pas besoin de configurer IAM des autorisations supplémentaires tant que vous avez le droit d'exécuter le document Automation et les actions appelées par le document. Si vous disposez d'autorisations d'administrateur dans IAM , vous avez déjà l'autorisation d'exécuter cette automatisation.

> **Note**
>
> Pour plus d'informations sur l'exécution d'une automatisation qui utilise un rôle de IAM service ou des formes plus avancées d'administration déléguée, consultez Exécution d'automatisations à l'aide de différents modèles de sécurité (p. 364) .

Exécution d'une automatisation étape par étape (console)

La procédure suivante montre comment utiliser la Systems Manager console pour exécuter manuellement une automatisation étape par étape.

Pour exécuter une automatisation étape par étape

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Automation (Automatisation), puis Execute automation (Exécuter l'automatisation).
3. Dans la liste Automation document (Document d'automatisation), choisissez un document. Choisissez une ou plusieurs options dans le volet Document categories (Catégories de documents) pour filtrer les documents SSM en fonction de leur objectif. Pour afficher un document dont vous êtes propriétaire, sélectionnez l’onglet Owned by me (M’appartenant). Pour afficher un document partagé avec votre compte, sélectionnez l’onglet Shared with me (Partagé avec moi). Pour afficher tous les documents, sélectionnez l’onglet All documents (Tous les documents).

   > **Note**
   >
   > Pour afficher des informations sur un document, sélectionnez son nom.

4. Dans la section Document details (Détails du document), vérifiez que l'option Document version (Version de document) correspond à la version que vous souhaitez exécuter. Le système inclut les options de version suivantes :

- Default version at runtime (Version par défaut lors de l'exécution) : choisissez cette option si le document d'automatisation est mis à jour régulièrement et qu'une nouvelle version par défaut est affectée.
- Latest version at runtime (Dernière version lors de l'exécution) : choisissez cette option si le document Automation est mis à jour régulièrement et que vous souhaitez exécuter la version la plus récemment mise à jour.
- 1 (Default) [1 (Par défaut)] : choisissez cette option pour exécuter la première version du document, qui est la version par défaut.

5. Choisissez Suivant.
6. Dans la section Execution Mode (Mode d'exécution), choisissez Manual execution (Exécution manuelle).
7. Dans la section Input parameters (Paramètres d'entrée), spécifiez les entrées obligatoires. Le cas échéant, vous pouvez choisir un rôle de service IAM dans la liste AutomationAssumeRole.
8. Sélectionnez Exécuter.
9. Choisissez Exécuter cette étape lorsque vous êtes prêt à démarrer la première étape de l'automatisation. L'automatisation passe à l'étape 1 et s'interrompt avant d'exécuter toutes les étapes suivantes spécifiées dans le document Automation que vous avez choisi à l'étape 3 de cette procédure. Si le document possède plusieurs étapes, vous devez sélectionner Execute this step (Exécuter cette étape) pour chaque étape afin que le flux de travail continue.

   Note

   La console affiche le statut de l'automatisation. Si l'automatisation ne parvient pas à exécuter une étape, consultez Dépannage de Systems Manager Automation (p. 676) .
10. Une fois que vous avez terminé toutes les étapes spécifiées dans le document d'Automation, choisissez Terminer et afficher les résultats pour terminer l'automatisation et afficher les résultats.

Exécution d'une automatisation étape par étape (ligne de commande)

La procédure suivante décrit comment utiliser l' AWS CLI (sous Linux ou Windows) ou Outils AWS pour PowerShell pour exécuter manuellement une automatisation étape par étape.

Pour exécuter une automatisation étape par étape

1. Installez et configurez l'AWS CLI ou Outils AWS pour PowerShell si vous ne l'avez pas déjà fait.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).
2. Exécutez la commande suivante pour démarrer une automatisation manuelle.

   Linux

   ```
   aws ssm start-automation-execution \
       --document-name DocumentName \
       --mode Interactive \
       --parameters ParametersRequiredByDocument
   ```

   Windows

   ```
   aws ssm start-automation-execution ^
       --document-name DocumentName ^
       --mode Interactive ^
       --parameters ParametersRequiredByDocument
   ```

PowerShell

```
Start-SSMAutomationExecution `
    -DocumentName DocumentName `
    -Mode Interactive `
    -Parameter ParametersRequiredByDocument
```

Voici un exemple d'utilisation du document `AWS-RestartEC2Instance` pour redémarrer l'instance EC2 spécifiée.

Linux

```
aws ssm start-automation-execution \
    --document-name "AWS-RestartEC2Instance" \
    --mode Interactive \
    --parameters "InstanceId=i-1234567890abcdef0"
```

Windows

```
aws ssm start-automation-execution ^
    --document-name "AWS-RestartEC2Instance" ^
    --mode Interactive ^
    --parameters "InstanceId=i-1234567890abcdef0"
```

PowerShell

```
Start-SSMAutomationExecution `
    -DocumentName AWS-RestartEC2Instance `
    -Mode Interactive
    -Parameter @{"InstanceId"="i-1234567890abcdef0"}
```

Le système renvoie des informations similaires à celles qui suivent.

Linux

```
{
    "AutomationExecutionId": "ba9cd881-1b36-4d31-a698-0123456789ab"
}
```

Windows

```
{
    "AutomationExecutionId": "ba9cd881-1b36-4d31-a698-0123456789ab"
}
```

PowerShell

```
27ba8174-59ae-4e13-8626-0123456789ab
```

3. Exécutez la commande suivante lorsque vous êtes prêt à démarrer la première étape de l'automatisation. L'automatisation passe à l'étape 1 et s'interrompt avant d'exécuter toutes les étapes suivantes spécifiées dans le document Automation que vous avez choisi à l'étape 1 de cette procédure. Si le document comporte plusieurs étapes, vous devez exécuter la commande suivante pour chaque étape pour que le flux de travail continue.

Linux

```
aws ssm send-automation-signal \
    --automation-execution-id ba9cd881-1b36-4d31-a698-0123456789ab \
    --signal-type StartStep \
    --payload StepName="stopInstances"
```

Windows

```
aws ssm send-automation-signal ^
    --automation-execution-id ba9cd881-1b36-4d31-a698-0123456789ab ^
    --signal-type StartStep ^
    --payload StepName="stopInstances"
```

PowerShell

```
Send-SSMAutomationSignal `
    -AutomationExecutionId 27ba8174-59ae-4e13-8626-0123456789ab `
    -SignalType StartStep
    -Payload @{"StepName"="stopInstances"}
```

Il n'y a pas de sortie si la commande réussit.

4. Exécutez la commande suivante pour récupérer le statut de chaque exécution d'étape dans l'automatisation.

Linux

```
aws ssm describe-automation-step-executions \
    --automation-execution-id ba9cd881-1b36-4d31-a698-0123456789ab
```

Windows

```
aws ssm describe-automation-step-executions ^
    --automation-execution-id ba9cd881-1b36-4d31-a698-0123456789ab
```

PowerShell

```
Get-SSMAutomationStepExecution `
    -AutomationExecutionId 27ba8174-59ae-4e13-8626-e177cdc11686
```

Le système renvoie des informations similaires à celles qui suivent.

Linux

```
{
    "StepExecutions": [
        {
            "StepName": "stopInstances",
            "Action": "aws:changeInstanceState",
            "ExecutionStartTime": 1557167178.42,
            "ExecutionEndTime": 1557167220.617,
            "StepStatus": "Success",
            "Inputs": {
                "DesiredState": "\"stopped\"",
```

```
                "InstanceIds": "[\"i-1234567890abcdef0\"]"
            },
            "Outputs": {
                "InstanceStates": [
                    "stopped"
                ]
            },
            "StepExecutionId": "654243ba-71e3-4771-b04f-0123456789ab",
            "OverriddenParameters": {},
            "ValidNextSteps": [
                "startInstances"
            ]
        },
        {
            "StepName": "startInstances",
            "Action": "aws:changeInstanceState",
            "ExecutionStartTime": 1557167273.754,
            "ExecutionEndTime": 1557167480.73,
            "StepStatus": "Success",
            "Inputs": {
                "DesiredState": "\"running\"",
                "InstanceIds": "[\"i-1234567890abcdef0\"]"
            },
            "Outputs": {
                "InstanceStates": [
                    "running"
                ]
            },
            "StepExecutionId": "8a4a1e0d-dc3e-4039-a599-0123456789ab",
            "OverriddenParameters": {}
        }
    ]
}
```

Windows

```
{
    "StepExecutions": [
        {
            "StepName": "stopInstances",
            "Action": "aws:changeInstanceState",
            "ExecutionStartTime": 1557167178.42,
            "ExecutionEndTime": 1557167220.617,
            "StepStatus": "Success",
            "Inputs": {
                "DesiredState": "\"stopped\"",
                "InstanceIds": "[\"i-1234567890abcdef0\"]"
            },
            "Outputs": {
                "InstanceStates": [
                    "stopped"
                ]
            },
            "StepExecutionId": "654243ba-71e3-4771-b04f-0123456789ab",
            "OverriddenParameters": {},
            "ValidNextSteps": [
                "startInstances"
            ]
        },
        {
            "StepName": "startInstances",
            "Action": "aws:changeInstanceState",
            "ExecutionStartTime": 1557167273.754,
            "ExecutionEndTime": 1557167480.73,
```

```
            "StepStatus": "Success",
            "Inputs": {
                "DesiredState": "\"running\"",
                "InstanceIds": "[\"i-1234567890abcdef0\"]"
            },
            "Outputs": {
                "InstanceStates": [
                    "running"
                ]
            },
            "StepExecutionId": "8a4a1e0d-dc3e-4039-a599-0123456789ab",
            "OverriddenParameters": {}
        }
    ]
}
```

PowerShell

```
Action               : aws:changeInstanceState
ExecutionEndTime     : 5/6/2019 19:45:46
ExecutionStartTime   : 5/6/2019 19:45:03
FailureDetails       :
FailureMessage       :
Inputs               : {[DesiredState, "stopped"], [InstanceIds,
 ["i-1234567890abcdef0"]]}
IsCritical           : False
IsEnd                : False
MaxAttempts          : 0
NextStep             :
OnFailure            :
Outputs              : {[InstanceStates,
 Amazon.Runtime.Internal.Util.AlwaysSendList`1[System.String]]}
OverriddenParameters : {}
Response             :
ResponseCode         :
StepExecutionId      : 8fcc9641-24b7-40b3-a9be-0123456789ab
StepName             : stopInstances
StepStatus           : Success
TimeoutSeconds       : 0
ValidNextSteps       : {startInstances}
```

5. Exécutez la commande suivante pour terminer l'automatisation une fois toutes les étapes spécifiées dans le document Automation choisi terminées.

   Linux

```
aws ssm stop-automation-execution \
    --automation-execution-id ba9cd881-1b36-4d31-a698-0123456789ab \
    --type Complete
```

   Windows

```
aws ssm stop-automation-execution ^
    --automation-execution-id ba9cd881-1b36-4d31-a698-0123456789ab ^
    --type Complete
```

   PowerShell

```
Stop-SSMAutomationExecution `
    -AutomationExecutionId 27ba8174-59ae-4e13-8626-0123456789ab `
```

```
    -Type Complete
```

Il n'y a pas de sortie si la commande réussit.

Exécution d'une automatisation avec des approbateurs

Les procédures suivantes décrivent comment utiliser la console AWS Systems Manager, l'AWS Command Line Interface (AWS CLI) et les Outils AWS pour Windows PowerShell pour exécuter un flux de travail AWS Systems Manager Automation avec des approbations à l'aide d'une exécution simple. Le flux de travail utilise l'action d'automatisation`aws:approve`, qui interrompt temporairement l'automatisation jusqu'à ce que les mandataires désignés approuvent ou refusent l'action. L'automatisation s'exécute dans le contexte de l'utilisateur AWS Identity and Access Management (IAM) actuel. Cela signifie que vous n'avez pas besoin de configurer IAM d'autorisations supplémentaires tant que vous avez le droit d'exécuter le document Automation et les actions appelées par le document. Si vous disposez d'autorisations d'administrateur dans IAM , vous avez déjà l'autorisation d'exécuter cette automatisation.

> **Note**
>
> Pour plus d'informations sur l'exécution d'une automatisation qui utilise un rôle de IAM service ou des formes plus avancées d'administration déléguée, consultez Exécution d'automatisations à l'aide de différents modèles de sécurité (p. 364) .

Avant de commencer

En plus des entrées standard requises par le document Automation, l'action `aws:approve` nécessite les deux paramètres suivants :

- Une liste d'approbateurs. La liste des approbateurs doit contenir au moins un approbateur sous la forme d'un utilisateur IAM ou d'un ARN d'utilisateur. Si plusieurs approbateurs sont fournis, un nombre minimum d'approbations correspondant doit également être spécifié dans le document Automation.
- Un ARN de rubrique Amazon Simple Notification Service (Amazon SNS). Le nom de la rubrique Amazon SNS doit commencer par `Automation`.

Cette procédure suppose que vous ayez déjà créé une rubrique Amazon SNS, ce qui est nécessaire pour diffuser la demande d'approbation. Pour de plus amples informations, veuillez consulter Création d'une rubrique dans le Amazon Simple Notification Service Manuel du développeur.

Exécution d'une automatisation avec des approbateurs (console)

Pour exécuter une automatisation avec des approbateurs

La procédure suivante décrit comment utiliser la Systems Manager console pour exécuter une automatisation avec des approbateurs.

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Automation (Automatisation), puis Execute automation (Exécuter l'automatisation).
3. Dans la liste Automation document (Document d'automatisation), choisissez un document. Choisissez une ou plusieurs options dans le volet Document categories (Catégories de documents) pour filtrer les documents SSM en fonction de leur objectif. Pour afficher un document dont vous êtes propriétaire, sélectionnez l’onglet Owned by me (M’appartenant). Pour afficher un document partagé avec votre compte, sélectionnez l’onglet Shared with me (Partagé avec moi). Pour afficher tous les documents, sélectionnez l’onglet All documents (Tous les documents).

Note

Pour afficher des informations sur un document, sélectionnez son nom.

4. Dans la section Document details (Détails du document), vérifiez que l'option Document version (Version de document) correspond à la version que vous souhaitez exécuter. Le système inclut les options de version suivantes :
   - Default version at runtime (Version par défaut lors de l'exécution) : choisissez cette option si le document d'automatisation est mis à jour régulièrement et qu'une nouvelle version par défaut est affectée.
   - Latest version at runtime (Dernière version lors de l'exécution) : choisissez cette option si le document Automation est mis à jour régulièrement et que vous souhaitez exécuter la version la plus récemment mise à jour.
   - 1 (Default) [1 (Par défaut)] : choisissez cette option pour exécuter la première version du document, qui est la version par défaut.
5. Choisissez Suivant.
6. Sur la page Execute automation document (Exécuter le document Automation), choisissez Simple execution (Exécution simple).
7. Dans la section Paramètres d'entrée, spécifiez les paramètres d'entrée obligatoires.

   Par exemple, si vous avez choisi le document AWS-StartEC2InstanceWithApproval, vous devez spécifier ou choisir une instance IDs pour le InstanceId paramètre.
8. Dans la section approbateurs, spécifiez les utilisateurs IAM ou IARNs'utilisateur des approbateurs pour l'action d'automatisation.
9. Dans la SNSTopicARN section, spécifiez l'ARN de la rubrique SNS à utiliser pour envoyer la notification d'approbation. Le nom de la rubrique SNS doit commencer par Automation.
10. Le cas échéant, vous pouvez choisir un rôle de IAM service dans la AutomationAssumeRole liste.
11. Choisissez Execute automation (Exécuter l'automatisation).

L'approbateur spécifié reçoit une Amazon SNS notification avec des détails pour approuver ou rejeter l'automatisation. Cette action d'approbation est valide pendant 7 jours à compter de la date de publication et peut être émise à l'aide de la console Systems Manager ou de l'AWS Command Line Interface (AWS CLI).

Si vous avez choisi d'approuver l'automatisation, le flux de travail continue d'exécuter les étapes incluses dans le document Automation spécifié. La console affiche le statut de l'automatisation. Si l'exécution de l'automatisation échoue, consultez Dépannage de Systems Manager Automation (p. 676) .

Pour approuver ou refuser une automatisation

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le volet de navigation, choisissez Automation, puis sélectionnez l'automatisation qui a été exécutée dans la procédure précédente.
3. Choisissez Actions puis Approve/Deny (Approuver/Refuser).
4. Choisissez d'Approuver ou de Refuser et, le cas échéant, saisissez un commentaire.
5. Choisissez Submit.

Exécution d'une automatisation avec des approbateurs (ligne de commande)

La procédure suivante décrit comment utiliser l' AWS CLI (sous Linux ou Windows) ou Outils AWS pour PowerShell pour exécuter une automatisation avec des approbateurs.

Pour exécuter une automatisation avec des approbateurs

1. Installez et configurez l'AWS CLI ou Outils AWS pour PowerShell si vous ne l'avez pas déjà fait.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Utilisez la commande suivante pour exécuter une automatisation avec des approbateurs. Dans la section Nom du document, spécifiez un document Automation qui inclut l'action d'automatisation`aws:approve`, .

   Pour `Approvers` , spécifiez les IAM utilisateurs ou utilisateurs ARNs d'approbateurs pour l'action. Pour `SNSTopic`, spécifiez l'ARN de la rubrique SNS à utiliser pour envoyer la notification d'approbation. Le nom de la rubrique SNS doit commencer par Automation.

   > **Note**
   >
   > Le nom exact des valeurs de paramètres pour les approbateurs et la rubrique SNS dépendent des valeurs spécifiées dans le document que vous choisissez.

   Linux

   ```
   aws ssm start-automation-execution \
       --document-name "AWS-StartEC2InstanceWithApproval" \
       --parameters
    "InstanceId=i-1234567890abcdef0,Approvers=arn:aws:iam::123456789012:role/
   Administrator,SNSTopicArn=arn:aws:sns:us-east-1:123456789012:AutomationApproval"
   ```

   Windows

   ```
   aws ssm start-automation-execution ^
       --document-name "AWS-StartEC2InstanceWithApproval" ^
       --parameters
    "InstanceId=i-1234567890abcdef0,Approvers=arn:aws:iam::123456789012:role/
   Administrator,SNSTopicArn=arn:aws:sns:us-east-1:123456789012:AutomationApproval"
   ```

   PowerShell

   ```
   Start-SSMAutomationExecution `
       -DocumentName AWS-StartEC2InstanceWithApproval `
       -Parameters @{
           "InstanceId"="i-1234567890abcdef0"
           "Approvers"="arn:aws:iam::123456789012:role/Administrator"
           "SNSTopicArn"="arn:aws:sns:us-east-1:123456789012:AutomationApproval"
       }
   ```

   Le système renvoie des informations similaires à celles qui suivent.

   Linux

   ```
   {
       "AutomationExecutionId": "df325c6d-b1b1-4aa0-8003-6cb7338213c6"
   }
   ```

   Windows

   ```
   {
       "AutomationExecutionId": "df325c6d-b1b1-4aa0-8003-6cb7338213c6"
   ```

```
}
```

PowerShell

```
462fa82a-7fff-430a-8490-0123456789ab
```

Pour approuver une automatisation

- Exécutez la commande suivante pour approuver une automatisation.

  Linux

```
aws ssm send-automation-signal \
    --automation-execution-id "4105a4fc-f944-11e6-9d32-0123456789ab" \
    --signal-type "Approve" \
    --payload "Comment=Replace_This_With_Approve_Comment"
```

  Windows

```
aws ssm send-automation-signal ^
    --automation-execution-id "4105a4fc-f944-11e6-9d32-0123456789ab" ^
    --signal-type "Approve" ^
    --payload "Comment=Replace_This_With_Approve_Comment"
```

  PowerShell

```
Send-SSMAutomationSignal `
    -AutomationExecutionId 462fa82a-7fff-430a-8490-0123456789ab `
    -SignalType Approve `
    -Payload @{"Comment"="Replace_This_With_Approval_Comment"}
```

  Il n'y a pas de sortie si la commande réussit.

Pour refuser une automatisation

- Exécutez la commande suivante pour refuser une automatisation.

  Linux

```
aws ssm send-automation-signal \
    --automation-execution-id "4105a4fc-f944-11e6-9d32-0123456789ab" \
    --signal-type "Deny" \
    --payload "Comment=Replace_This_With_Deny_Comment"
```

  Windows

```
aws ssm send-automation-signal ^
    --automation-execution-id "4105a4fc-f944-11e6-9d32-0123456789ab" ^
    --signal-type "Deny" ^
    --payload "Comment=Replace_This_With_Deny_Comment"
```

PowerShell

```
Send-SSMAutomationSignal `
    -AutomationExecutionId 462fa82a-7fff-430a-8490-0123456789ab `
    -SignalType Deny `
    -Payload @{"Comment"="Replace_This_With_Deny_Comment"}
```

Il n'y a pas de sortie si la commande réussit.

Exécution d'automatisations qui utilisent des cibles et des contrôles du débit

vous AWS Systems Manager permet d'exécuter des automatisations sur un parc de AWS ressources à l'aide de cibles. En outre, vous pouvez contrôler le déploiement de l'automatisation dans l'ensemble de votre parc en spécifiant une valeur de simultanéité et un seuil d'erreurs. La valeur de simultanéité détermine le nombre de ressources autorisées à exécuter l'automatisation simultanément. Un seuil d'erreurs détermine le nombre d'automatisations autorisées à échouer avant qu' ne Systems Manager cesse d'envoyer le flux de travail à d'autres ressources. Les fonctions de simultanéité et de seuil d'erreurs sont appelées collectivement contrôles du débit.

Pour de plus amples informations sur la simultanéité et les seuils d'erreurs, veuillez consulter À propos de la simultanéité et des seuils d'erreur (p. 343). Pour de plus amples informations sur les cibles, veuillez consulter À propos des cibles (p. 337).

Les procédures suivantes décrivent comment exécuter une automatisation avec des cibles et des contrôles du débit à l'aide de la AWS Systems Manager console , de l' AWS Command Line Interface (AWS CLI) et des Outils AWS pour Windows PowerShell .

Exécution d'une automatisation avec des cibles et des contrôles du débit (console)

La procédure suivante décrit comment utiliser la Systems Manager console pour exécuter une automatisation avec des cibles et des contrôles du débit.

Pour exécuter une automatisation avec des cibles et des contrôles du débit

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Automation (Automatisation), puis Execute automation (Exécuter l'automatisation).
3. Dans la liste Automation document (Document d'automatisation), choisissez un document. Choisissez une ou plusieurs options dans le volet Document categories (Catégories de documents) pour filtrer les documents SSM en fonction de leur objectif. Pour afficher un document dont vous êtes propriétaire, sélectionnez l’onglet Owned by me (M’appartenant). Pour afficher un document partagé avec votre compte, sélectionnez l’onglet Shared with me (Partagé avec moi). Pour afficher tous les documents, sélectionnez l’onglet All documents (Tous les documents).

   Note

   Pour afficher des informations sur un document, sélectionnez son nom.
4. Dans la section Document details (Détails du document), vérifiez que l'option Document version (Version de document) correspond à la version que vous souhaitez exécuter. Le système inclut les options de version suivantes :

- Default version at runtime (Version par défaut lors de l'exécution) : choisissez cette option si le document d'automatisation est mis à jour régulièrement et qu'une nouvelle version par défaut est affectée.
- Latest version at runtime (Dernière version lors de l'exécution) : choisissez cette option si le document Automation est mis à jour régulièrement et que vous souhaitez exécuter la version la plus récemment mise à jour.
- 1 (Default) [1 (Par défaut)] : choisissez cette option pour exécuter la première version du document, qui est la version par défaut.

5. Choisissez Suivant.
6. Dans la section Execution Mode (Mode d'exécution), choisissez Rate Control (Contrôle du débit). Vous devez utiliser ce mode ou Multi-account and Region (Compte et région multiples) si vous souhaitez utiliser des cibles et des contrôles du débit.
7. Dans la section Targets (Cibles), choisissez la façon dont vous souhaitez cibler les ressources AWS où vous souhaitez exécuter l'automatisation. Ces options sont obligatoires.
   a. Utilisez la liste Parameter (Paramètre) pour choisir un paramètre. Les éléments de la liste Parameter (Paramètre) sont déterminés par les paramètres du document d'automatisation que vous avez sélectionnés au début de cette procédure. En choisissant un paramètre, vous définissez le type de ressource sur lequel le flux de travail d'automatisation s'exécutera.
   b. Utilisez la liste Targets (Cibles) pour choisir la façon dont vous souhaitez cibler les ressources.
      i. Si vous avez choisi de cibler des ressources à l'aide de valeurs de paramètre, entrez la valeur de paramètre que vous avez sélectionnée dans la section Input parameters (Paramètres d'entrée).
      ii. Si vous choisissez de cibler les ressources à l'aide de Groupes de ressources AWS, choisissez le nom du groupe dans la liste Resource Group (Groupe de ressources).
      iii. Si vous choisissez de cibler des ressources à l'aide de balises, entrez la clé de balise et (éventuellement) la valeur de balise dans les champs fournis. Choisissez Ajouter.
      iv. Si vous voulez exécuter un playbook d'automatsation sur toutes les instances du compte AWS actuel et de la région, alors sélectionnez All instances (Toutes les instances).
8. Dans la section Input parameters (Paramètres d'entrée), spécifiez les entrées obligatoires. Le cas échéant, vous pouvez choisir un rôle de service IAM dans la liste AutomationAssumeRole.

   Note

   Vous n'aurez peut-être pas besoin de choisir certaines options dans la section Input parameters (Paramètres d'entrée). Cela est dû au fait que vous avez ciblé des ressources à l'aide de balises ou d'un groupe de ressources. Par exemple, si vous avez choisi le document AWS-RestartEC2Instance, vous n'avez pas à spécifier ou à choisir d'ID d'instance dans la section Input parameters (Paramètres d'entrée). L'exécution d'automatisation localise les instances à redémarrer en utilisant les balises ou le groupe de ressources que vous avez spécifiés.
9. Utilisez les options de la section Rate control (Contrôle du débit) pour restreindre le nombre de ressources AWS qui peuvent exécuter l'automatisation au sein de chaque paire compte-région.

   Dans la section Concurrency (Simultanéité), choisissez une option :

   - Choisissez targets (cibles) pour entrer un nombre absolu de cibles pouvant exécuter le flux de travail Automation simultanément.
   - Choisissez percentage (pourcentage) pour indiquer un pourcentage de l'ensemble de cibles pouvant exécuter le flux de travail Automation simultanément.
10. Dans la section Error threshold (Seuil d'erreurs), choisissez une option :

- Choisissez errors (erreurs) pour indiquer un nombre absolu d'erreurs autorisées avant que l'automatisation ne cesse d'envoyer le flux de travail à d'autres ressources.
- Choisissez percentage (pourcentage) pour indiquer un pourcentage d'erreurs autorisées avant que l'automatisation ne cesse d'envoyer le flux de travail à d'autres ressources.

11. Sélectionnez Exécuter.

Pour afficher les automatisations lancées par votre automatisation de contrôle de débit, dans le volet de navigation, choisissez Automation, puis sélectionnez Show child automations (Afficher les automatisations enfants).

Exécution d'une automatisation avec des cibles et des contrôles du débit (ligne de commande)

La procédure suivante décrit comment utiliser l' AWS CLI (sous Linux ou Windows) ou Outils AWS pour PowerShell pour exécuter une automatisation avec des cibles et des contrôles du débit.

Pour exécuter une automatisation avec des cibles et des contrôles du débit

1. Installez et configurez l'AWS CLI ou Outils AWS pour PowerShell si vous ne l'avez pas déjà fait.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Exécutez la commande suivante pour afficher une liste de documents.

   Linux

   ```
   aws ssm list-documents
   ```

   Windows

   ```
   aws ssm list-documents
   ```

   PowerShell

   ```
   Get-SSMDocumentList
   ```

   Notez le nom du document Automation que vous souhaitez exécuter.

3. Exécutez la commande suivante pour afficher des informations détaillées relatives au document Automation. Notez un nom de paramètre (par exemple, `InstanceId`) que vous souhaitez utiliser pour l'option `--target-parameter-name`. Ce paramètre détermine le type de ressource sur lequel l'automatisation s'exécute.

   Linux

   ```
   aws ssm describe-document \
       --name document_name
   ```

   Windows

   ```
   aws ssm describe-document ^
       --name document_name
   ```

PowerShell

```
Get-SSMDocumentDescription `
    -Name document_name
```

4. Créez une commande qui utilise les options de cibles et de contrôle du débit que vous souhaitez exécuter. Pour vous aider, voici quelques modèles de commandes.

Ciblage à l'aide de balises

Linux

```
aws ssm start-automation-execution \
    --document-name document_name \
    --targets Key=tag:key_name,Values=value \
    --target-parameter-name parameter_name \
    --parameters
 "input_parameter_name1=input_parameter_value1,input_parameter_name2=input_parameter_value2"
 \
    --max-concurrency 10 \
    --max-errors 25%
```

Windows

```
aws ssm start-automation-execution ^
    --document-name document_name ^
    --targets Key=tag:key_name,Values=value ^
    --target-parameter-name parameter_name ^
    --parameters
 "input_parameter_name1=input_parameter_value1,input_parameter_name2=input_parameter_value2"
 ^
    --max-concurrency 10 ^
    --max-errors 25%
```

PowerShell

```
$Targets = New-Object Amazon.SimpleSystemsManagement.Model.Target
$Targets.Key = "tag:key_name"
$Targets.Values = "value"

Start-SSMAutomationExecution `
    DocumentName "DocumentName" `
    -Targets $Targets `
    -TargetParameterName "Parameter_Name" `
    -Parameter
 @{"input_parameter_name1"="input_parameter_value1";"input_parameter_name2"="input_parameter_va
 `
    -MaxConcurrency "a_number_of_instances_or_a_percentage_of_target_set" `
    -MaxError "a_number_of_errors_or_a_percentage_of_target_set"
```

Ciblage à l'aide des valeurs de paramètre

Linux

```
aws ssm start-automation-execution \
    --document-name document_name \
    --targets Key=ParameterValues,Values=value_1,value_2,value_3 \
    --target-parameter-name parameter_name \
```

```
    --parameters "input_parameter_name1=input_parameter_value1" \
    --max-concurrency 10 \
    --max-errors 25%
```

Windows

```
aws ssm start-automation-execution ^
    --document-name document_name ^
    --targets Key=ParameterValues,Values=value_1,value_2,value_3 ^
    --target-parameter-name parameter_name ^
    --parameters "input_parameter_name1=input_parameter_value1" ^
    --max-concurrency 10 ^
    --max-errors 25%
```

PowerShell

```
$Targets = New-Object Amazon.SimpleSystemsManagement.Model.Target
$Targets.Key = "ParameterValues"
$Targets.Values = "value_1","value_2","value_3"

Start-SSMAutomationExecution `
    -DocumentName "DocumentName" `
    -Targets $Targets `
    -TargetParameterName "Parameter_Name" `
    -Parameter @{"input_parameter_name1"="input_parameter_value1"} `
    -MaxConcurrency "a_number_of_instances_or_a_percentage_of_target_set" `
    -MaxError "a_number_of_errors_or_a_percentage_of_target_set"
```

Ciblage à l'aide de Groupes de ressources AWS

Linux

```
aws ssm start-automation-execution \
    --document-name document_name \
    --targets Key=ResourceGroup,Values=Resource_Group_name \
    --target-parameter-name parameter_name \
    --parameters
 "input_parameter_name1=input_parameter_value1,input_parameter_name2=input_parameter_value2"
 \
    --max-concurrency 10 \
    --max-errors 25%
```

Windows

```
aws ssm start-automation-execution ^
    --document-name document_name ^
    --targets Key=ResourceGroup,Values=Resource_Group_name ^
    --target-parameter-name parameter_name ^
    --parameters
 "input_parameter_name1=input_parameter_value1,input_parameter_name2=input_parameter_value2"
 ^
    --max-concurrency 10 ^
    --max-errors 25%
```

PowerShell

```
$Targets = New-Object Amazon.SimpleSystemsManagement.Model.Target
$Targets.Key = "ResourceGroup"
```

```
$Targets.Values = "Resource_Group_Name"

Start-SSMAutomationExecution `
    -DocumentName "DocumentName" `
    -Targets $Targets `
    -TargetParameterName "Parameter_Name" `
    -Parameter
 @{"input_parameter_name1"="input_parameter_value1";"input_parameter_name2"="input_parameter_va
 `
    -MaxConcurrency "a_number_of_instances_or_a_percentage_of_target_set" `
    -MaxError "a_number_of_errors_or_a_percentage_of_target_set"
```

Ciblage de toutes les instances du compte et de la région AWS actuels

Linux

```
aws ssm start-automation-execution \
    --document-name document_name \
    --targets "Key=AWS::EC2::Instance,Values=*"  \
    --target-parameter-name instanceId \
    --parameters "input_parameter_name1=input_parameter_value1" \
    --max-concurrency 10 \
    --max-errors 25%
```

Windows

```
aws ssm start-automation-execution ^
    --document-name document_name ^
    --targets Key=AWS::EC2::Instance,Values=* ^
    --target-parameter-name instanceId ^
    --parameters "input_parameter_name1=input_parameter_value1" ^
    --max-concurrency 10 ^
    --max-errors 25%
```

PowerShell

```
$Targets = New-Object Amazon.SimpleSystemsManagement.Model.Target
$Targets.Key = "AWS::EC2::Instance"
$Targets.Values = "*"

Start-SSMAutomationExecution `
    -DocumentName "DocumentName" `
    -Targets $Targets `
    -TargetParameterName "instanceId" `
    -Parameter @{"input_parameter_name1"="input_parameter_value1"} `
    -MaxConcurrency "a_number_of_instances_or_a_percentage_of_target_set" `
    -MaxError "a_number_of_errors_or_a_percentage_of_target_set"
```

La commande renvoie un ID d'exécution. Copiez cet ID dans le Presse-papiers. Vous pouvez utiliser cet ID pour afficher le statut de l'automatisation.

Linux

```
{
    "AutomationExecutionId": "a4a3c0e9-7efd-462a-8594-01234EXAMPLE"
}
```

Windows

```
{
    "AutomationExecutionId": "a4a3c0e9-7efd-462a-8594-01234EXAMPLE"
}
```

PowerShell

```
a4a3c0e9-7efd-462a-8594-01234EXAMPLE
```

5. Exécutez la commande suivante pour afficher l'exécution de l'automatisation.

   Linux

```
aws ssm describe-automation-executions \
    --filter Key=ExecutionId,Values=a4a3c0e9-7efd-462a-8594-01234EXAMPLE
```

   Windows

```
aws ssm describe-automation-executions ^
    --filter Key=ExecutionId,Values=a4a3c0e9-7efd-462a-8594-01234EXAMPLE
```

   PowerShell

```
Get-SSMAutomationExecutionList | `
    Where {$_.AutomationExecutionId -eq "a4a3c0e9-7efd-462a-8594-01234EXAMPLE"}
```

6. Pour afficher les détails relatifs à la progression de l'automatisation, exécutez la commande suivante.

   Linux

```
aws ssm get-automation-execution \
    --automation-execution-id a4a3c0e9-7efd-462a-8594-01234EXAMPLE
```

   Windows

```
aws ssm get-automation-execution ^
    --automation-execution-id a4a3c0e9-7efd-462a-8594-01234EXAMPLE
```

   PowerShell

```
Get-SSMAutomationExecution `
    -AutomationExecutionId a4a3c0e9-7efd-462a-8594-01234EXAMPLE
```

   Le système renvoie des informations similaires à celles qui suivent.

   Linux

```
{
    "AutomationExecution": {
        "StepExecutionsTruncated": false,
        "AutomationExecutionStatus": "Success",
        "MaxConcurrency": "1",
        "Parameters": {},
```

```
        "MaxErrors": "1",
        "Outputs": {},
        "DocumentName": "AWS-StopEC2Instance",
        "AutomationExecutionId": "a4a3c0e9-7efd-462a-8594-01234EXAMPLE",
        "ResolvedTargets": {
            "ParameterValues": [
                "i-02573cafcfEXAMPLE"
            ],
            "Truncated": false
        },
        "ExecutionEndTime": 1564681619.915,
        "Targets": [
            {
                "Values": [
                    "DEV"
                ],
                "Key": "tag:ENV"
            }
        ],
        "DocumentVersion": "1",
        "ExecutionStartTime": 1564681576.09,
        "ExecutedBy": "arn:aws:sts::123456789012:assumed-role/Administrator/Admin",
        "StepExecutions": [
            {
                "Inputs": {
                    "InstanceId": "i-02573cafcfEXAMPLE"
                },
                "Outputs": {},
                "StepName": "i-02573cafcfEXAMPLE",
                "ExecutionEndTime": 1564681619.093,
                "StepExecutionId": "86c7b811-3896-4b78-b897-01234EXAMPLE",
                "ExecutionStartTime": 1564681576.836,
                "Action": "aws:executeAutomation",
                "StepStatus": "Success"
            }
        ],
        "TargetParameterName": "InstanceId",
        "Mode": "Auto"
    }
}
```

Windows

```
{
    "AutomationExecution": {
        "StepExecutionsTruncated": false,
        "AutomationExecutionStatus": "Success",
        "MaxConcurrency": "1",
        "Parameters": {},
        "MaxErrors": "1",
        "Outputs": {},
        "DocumentName": "AWS-StopEC2Instance",
        "AutomationExecutionId": "a4a3c0e9-7efd-462a-8594-01234EXAMPLE",
        "ResolvedTargets": {
            "ParameterValues": [
                "i-02573cafcfEXAMPLE"
            ],
            "Truncated": false
        },
        "ExecutionEndTime": 1564681619.915,
        "Targets": [
            {
                "Values": [
                    "DEV"
```

```
                ],
                "Key": "tag:ENV"
            }
        ],
        "DocumentVersion": "1",
        "ExecutionStartTime": 1564681576.09,
        "ExecutedBy": "arn:aws:sts::123456789012:assumed-role/Administrator/Admin",
        "StepExecutions": [
            {
                "Inputs": {
                    "InstanceId": "i-02573cafcfEXAMPLE"
                },
                "Outputs": {},
                "StepName": "i-02573cafcfEXAMPLE",
                "ExecutionEndTime": 1564681619.093,
                "StepExecutionId": "86c7b811-3896-4b78-b897-01234EXAMPLE",
                "ExecutionStartTime": 1564681576.836,
                "Action": "aws:executeAutomation",
                "StepStatus": "Success"
            }
        ],
        "TargetParameterName": "InstanceId",
        "Mode": "Auto"
    }
}
```

PowerShell

```
AutomationExecutionId       : a4a3c0e9-7efd-462a-8594-01234EXAMPLE
AutomationExecutionStatus   : Success
CurrentAction               :
CurrentStepName             :
DocumentName                : AWS-StopEC2Instance
DocumentVersion             : 1
ExecutedBy                  : arn:aws:sts::123456789012:assumed-role/Administrator/
Admin
ExecutionEndTime            : 8/1/2019 5:46:59 PM
ExecutionStartTime          : 8/1/2019 5:46:16 PM
FailureMessage              :
MaxConcurrency              : 1
MaxErrors                   : 1
Mode                        : Auto
Outputs                     : {}
Parameters                  : {}
ParentAutomationExecutionId :
ProgressCounters            :
ResolvedTargets             : Amazon.SimpleSystemsManagement.Model.ResolvedTargets
StepExecutions              : {i-02573cafcfEXAMPLE}
StepExecutionsTruncated     : False
Target                      :
TargetLocations             : {}
TargetMaps                  : {}
TargetParameterName         : InstanceId
Targets                     : {tag:Name}
```

Note

Vous pouvez également surveiller le statut de l'automatisation dans la console. Dans la liste des exécutions d'Automation, choisissez l'automatisation que vous venez d'exécuter, puis choisissez l'onglet Execution steps (Étapes d'exécution). Cet onglet affiche le statut des actions d'automatisation.

À propos des cibles

Le `Targets` paramètre vous permet de définir rapidement les ressources de votre parc qui peuvent exécuter une automatisation. Par exemple, si vous souhaitez exécuter une automatisation qui redémarre vos instances gérées, au lieu de sélectionner manuellement des dizaines d'instances IDs dans la console ou de les saisir dans une commande, vous pouvez cibler les instances en spécifiant des balises EC2 avec le `Targets` paramètre .

Lorsque vous exécutez une automatisation qui utilise une cible, Systems Manager crée une automatisation enfant pour chaque cible. Par exemple, si vous ciblez des volumes Amazon Elastic Block Store (Amazon EBS) en spécifiant des balises, et que ces balises sont résolues en 100 Amazon EBS volumes, Systems Manager crée 100 flux de travail d'automatisation enfant. L'automatisation parent est terminée lorsque toutes les automatisations enfants atteignent un état final.

> **Note**
>
> Tous ceux `input parameters` que vous spécifiez lors de l'exécution (dans la section Paramètres d'entrée de la console ou à l'aide de l'`parameters`option à partir de la ligne de commande) sont automatiquement traités par toutes les automatisations enfants.

Vous pouvez cibler des ressources pour une automatisation en utilisant des balises, Resource Groups des et des valeurs de paramètre. De plus, vous pouvez utiliser l'option `TargetMaps` pour cibler plusieurs valeurs de paramètre à l'aide de l'interface de ligne de commande ou d'un fichier. La section suivante décrit chacune de ces options de ciblage en détail.

Ciblage des balises

De nombreuses ressources AWS prennent en charge les balises, y compris les instances Amazon EC2 et Amazon Relational Database Service (Amazon RDS), les volumes et les instantanés Amazon Elastic Block Store (Amazon EBS), ainsi que les compartiments Resource Groups et Amazon Simple Storage Service (Amazon S3), pour n'en citer que quelques-uns. Vous pouvez rapidement exécuter l'automatisation sur vos AWS ressources en ciblant les balises. Une balise est une paire clé-valeur, telle que Système_Exploitation-Linux ou Service-Finances. Si vous affectez un nom spécifique à une ressource, vous pouvez également utiliser le mot « Nom » en tant que clé et le nom de la ressource comme valeur.

Lorsque vous spécifiez une balise comme cible d'une automatisation, vous spécifiez également un paramètre cible. Le paramètre cible utilise l'option `TargetParameterName`. En choisissant un paramètre cible, vous définissez le type de ressource sur lequel l'automatisation s'exécute. Le paramètre cible que vous spécifiez avec la balise doit être un paramètre valide défini dans le document d'Automation. Par exemple, si vous souhaitez cibler des dizaines d'instances EC2 avec des balises, choisissez le paramètre cible `InstanceId`. En choisissant ce paramètre, vous définissez les instances en tant que type de ressource pour l'automatisation. La capture d'écran suivante utilise le document AWS-DetachEBSVolume. Le paramètre cible logique est `VolumeId`.

Le document AWS-DetachEBSVolume inclut également une propriété spéciale appelée Type de cible, qui est définie sur `/AWS::EC2::Volume` . Cela signifie que si la paire balise-clé `Finance-TestEnv` renvoie différents types de ressources (par exemple, des instances EC2, Amazon EBS des volumes, des Amazon EBS instantanés), seuls les Amazon EBS volumes seront utilisés.

> **Important**
>
> Les noms de paramètre cible sont sensibles à la casse. Si vous exécutez des automatisations à l'aide de la AWS CLI ou Outils AWS pour Windows PowerShell de l' , vous devez entrer le nom du paramètre cible exactement tel qu'il est défini dans le document d'automatisation. Dans le cas contraire, le système renvoie une erreur `InvalidAutomationExecutionParametersException`. Vous pouvez utiliser l'action DescribeDocument d'API pour afficher des informations sur les paramètres cibles disponibles dans un document spécifique. Voici un exemple de commande AWS CLI qui fournit des informations sur le document AWS-DeleteSnapshot :
>
> ```
> aws ssm describe-document \
>  --name AWS-DeleteSnapshot
> ```

Voici quelques exemples de commandes AWS CLI qui ciblent des ressources en utilisant des balises.

Exemple 1 : Ciblage des balises à l'aide d'une paire clé-valeur pour redémarrer les instances EC2

Cet exemple redémarre toutes les instances EC2 qui sont balisées avec la clé Department et la valeur HumanResources . Le paramètre cible utilise le InstanceId paramètre du document d'Automation. L'exemple utilise un paramètre supplémentaire pour exécuter Automation à l'aide d'un rôle de service Automation (également appelé rôle responsable).

```
aws ssm start-automation-execution \
    --document-name AWS-RestartEC2Instance \
    --targets Key=tag:Department,Values=HumanResources \
    --target-parameter-name InstanceId \
    --parameters "AutomationAssumeRole=arn:aws:iam::111122223333:role/
AutomationServiceRole"
```

Exemple 2 : Ciblage de balises à l'aide d'une paire clé-valeur pour supprimer des Amazon EBS instantanés

L'exemple suivant utilise le document d'Automation AWS-DeleteSnapshot pour supprimer tous les instantanés avec une clé Name et une valeur .January2018Backups Le paramètre cible utilise le VolumeId paramètre.

```
aws ssm start-automation-execution \
    --document-name AWS-DeleteSnapshot \
    --targets Key=tag:Name,Values=January2018Backups \
    --target-parameter-name VolumeId
```

Ciblage des Groupes de ressources AWS

Vous pouvez spécifier un seul groupe de AWS ressources en tant que cible d'une automatisation. Systems Manager crée une automatisation enfant pour chaque objet dans le groupe de ressources cible.

Par exemple, supposons que l'un de vos groupes de ressources se nomme PatchedAMIs . Ce groupe de ressources inclut une liste de 25 Amazon Machine Images (AMI) Windows auxquelles des correctifs sont appliqués régulièrement. Si vous exécutez une automatisation qui utilise le document AWS-CreateManagedWindowsInstance et qui cible ce groupe de ressources, Systems Manager crée une automatisation enfant pour chacune des 25 AMIs . Cela signifie qu'en ciblant le groupe de PatchedAMIs ressources, l'automatisation crée 25 instances à partir d'une liste de corrigéesAMIs. L'automatisation parent est terminée lorsque toutes les automatisations enfants terminent le traitement ou atteignent un état final.

La AWS CLI commande suivante s'applique à l'exemple de groupe de PatchAMIs ressources. La commande prend le AmiId paramètre pour l'option --target-parameter-name. Cette commande n'inclut pas d'autres paramètres pour définir le type d'instance à créer à partir de chaque AMI. Le document AWS-CreateManagedWindowsInstance renvoie par défaut au type d'instance t2.medium. Dans ce cas, la commande créerait donc 25 instances EC2 t2.medium pour Windows Server.

```
aws ssm start-automation-execution \
    --document-name AWS-CreateManagedWindowsInstance \
    --targets Key=ResourceGroup,Values=PatchedAMIs  \
    --target-parameter-name AmiId
```

L'exemple de console suivant utilise un groupe de ressources appelé t2-micro-instances.

Ciblage des paramètres de valeur

Vous pouvez également cibler une valeur de paramètre. Vous entrez `ParameterValues` en tant que clé, puis entrez la valeur de ressource spécifique où vous souhaitez que l'automatisation s'exécute. Si vous spécifiez plusieurs valeurs, Systems Manager exécute une automatisation enfant sur chaque valeur spécifiée.

Par exemple, supposons que votre document Automation inclut un InstanceID paramètre . Si vous ciblez les valeurs du InstanceID paramètre lorsque vous exécutez Automation, Systems Manager exécute une automatisation enfant pour chaque valeur d'ID d'instance spécifiée. L'automatisation parent est terminée lorsque l'automatisation termine l'exécution de chaque instance spécifiée ou si l'exécution échoue. Vous pouvez cibler un maximum de 50 valeurs de paramètre.

L'exemple suivant utilise le document d'Automation AWS-CreateImage. Le nom du paramètre cible spécifié est InstanceId . La clé utilise ParameterValues . Les valeurs sont deux instances EC2.IDs Cette commande crée une automatisation pour chaque instance, ce qui génère une AMI à partir de chaque instance.

```
aws ssm start-automation-execution
    --document-name AWS-CreateImage \
    --target-parameter-name InstanceId \
    --targets Key=ParameterValues,Values=i-02573cafcfEXAMPLE,i-0471e04240EXAMPLE
```

> **Note**
>
> nAutomationAssumeRole'est pas un paramètre valide. Ne choisissez pas cet élément lors de l'exécution de flux de travail Automation qui ciblent une valeur de paramètre.

Ciblage des mappages de valeurs des paramètres

L`TargetMaps`'option étend votre capacité à cibler `ParameterValues` . Vous pouvez entrer un tableau de valeurs de paramètres en utilisant `TargetMaps` à la ligne de commande. Vous pouvez spécifier un maximum de 50 valeurs de paramètre dans la ligne de commande. Si vous souhaitez exécuter des commandes qui spécifient plus de 50 valeurs de paramètres, vous pouvez saisir ces valeurs dans un fichier JSON. Vous pouvez ensuite appeler ce fichier à partir de la ligne de commande.

> **Note**
>
> Les `TargetMaps` ne sont pas pris en charge dans la console.

Utilisez le format suivant pour spécifier plusieurs valeurs de paramètre en utilisant l'option `TargetMaps` dans une commande :

```
aws ssm start-automation-execution \
    --document-name name_of_document \
    --target-maps “parameterA=parameterA1, parameterB=parameterB1, parameterC=parameterC1”
 “parameterA=parameterA2, parameterB=parameterB2, parameterC=parameterC2”
 “parameterA=parameterA3, parameterB=parameterB3, parameterC=parameterC3”
```

Si vous souhaitez saisir plus de 50 valeurs de paramètres pour l'option `TargetMaps`, spécifiez les valeurs dans un fichier à l'aide du format JSON suivant. L'utilisation d'un fichier JSON améliore également la lisibilité lorsque vous devez fournir plusieurs valeurs de paramètre.

```
[

      {“parameterA”:”parameterValueA1”, “parameterB”:”parameterValueB1”,
 “parameterC”:”parameterValueC1”},

      {“parameterA”:”parameterValueA2”, “parameterB”:”parameterValueB2”,
 “parameterC”:”parameterValueC2”},
```

```
      {“parameterA”:”parameterValueA3”, “parameterB”:”parameterValueB3”,
 “parameterC”:”parameterValueC3”}

   ]
```

Enregistrez le fichier avec l'extension de fichier .json. Vous pouvez appeler le fichier en utilisant la commande suivante :

```
aws ssm start-automation-execution \
    --document-name name_of_document \
    --parameters one_or_more_input_parameters \
    --target-maps full_path_to_file/file_name.json
```

Vous pouvez également télécharger le fichier à partir d'un compartiment S3, dans la mesure où vous avez l'autorisation de lire les données de ce compartiment. Utilisez le format de commande suivant :

```
aws ssm start-automation-execution \
    --document-name name_of_document \
    --target-maps http://DOC-EXAMPLE-BUCKET.s3.amazonaws.com/file_name.json
```

Voici un exemple de scénario qui peut vous aider à comprendre l'option `TargetMaps`. Dans ce scénario, un utilisateur souhaite créer des instances EC2 de différents types à partir de .AMIs Pour effectuer cette tâche, l'utilisateur crée un document Automation nommé AMI_Testing. Ce document définit deux paramètres d'entrée : `instanceTypeet` `imageId` .

```
{
  "description": "AMI Testing",
  "schemaVersion": "0.3",
  "assumeRole": "{{assumeRole}}",
  "parameters": {
    "assumeRole": {
      "type": "String",
      "description": "Role under which to run the automation",
      "default": ""
    },
    "instanceType": {
      "type": "String",
      "description": "Type of EC2 Instance to launch for this test"
    },
    "imageId": {
      "type": "String",
      "description": "Source AMI id from which to run instance"
    }
  },
  "mainSteps": [
    {
      "name": "runInstances",
      "action": "aws:runInstances",
      "maxAttempts": 1,
      "onFailure": "Abort",
      "inputs": {
        "ImageId": "{{imageId}}",
        "InstanceType": "{{instanceType}}",
        "MinInstanceCount": 1,
        "MaxInstanceCount": 1
      }
    }
  ],
  "outputs": [
    "runInstances.InstanceIds"
  ]
```

```
}
```

L'utilisateur spécifie ensuite les valeurs de paramètres cibles suivantes dans un fichier nommé AMI_instance_types.json.

```
[
  {
    "instanceType" : ["t2.micro"],
    "imageId" : ["ami-b70554c8"]
  },
  {
    "instanceType" : ["t2.small"],
    "imageId" : ["ami-b70554c8"]
  },
  {
    "instanceType" : ["t2.medium"],
    "imageId" : ["ami-cfe4b2b0"]
  },
  {
    "instanceType" : ["t2.medium"],
    "imageId" : ["ami-cfe4b2b0"]
  },
  {
    "instanceType" : ["t2.medium"],
    "imageId" : ["ami-cfe4b2b0"]
  }
]
```

L'utilisateur peut exécuter l'automatisation et créer les cinq instances EC2 définies dans AMI_instance_types.json en exécutant la commande suivante :

```
aws ssm start-automation-execution \
    --document-name AMI_Testing \
    --target-parameter-name imageId \
    --target-maps file:///home/TestUser/workspace/runinstances/AMI_instance_types.json
```

Ciblage de toutes les instances

Vous pouvez exécuter une automatisation sur toutes les instances gérées dans le AWS compte et la région actuels en choisissant toutes les instances dans la liste des cibles. Par exemple, si vous souhaitez redémarrer toutes les instances gérées de votre AWS compte et de la région actuelle, vous pouvez choisir le document AWS-RestartEC2Instance, puis toutes les instances dans la liste des cibles.

Une fois que vous avez choisi All instances (Toutes les instances), remplit le Systems Manager champ Instance avec un astérisque (*) et rend le champ indisponible pour les modifications (le champ est

grisé). En outreSystems Manager, le InstanceId champ du champ Paramètres d'entrée n'est pas disponible pour les modifications. Rendre ces champs indisponibles pour empêcher leur modification est le comportement attendu si vous choisissez de cibler toutes les instances.

À propos de la simultanéité et des seuils d'erreur

Vous pouvez contrôler le déploiement d'une automatisation à travers une flotte de AWS ressources en spécifiant une valeur de simultanéité et un seuil d'erreurs. La simultanéité et le seuil d'erreurs sont appelés collectivement contrôles du débit.

Concurrency

La simultanéité vous permet de spécifier le nombre de ressources autorisées à exécuter une automatisation simultanément. La simultanéité permet de limiter l'impact ou les temps d'arrêt sur vos ressources lors du traitement d'une automatisation. Vous pouvez spécifier un nombre absolu de ressources (par exemple, 20) ou un pourcentage de l'ensemble de la cible (par exemple, 10 %).

Le système de mise en file d'attente transmet l'automatisation à une seule ressource et attend la fin de l'appel initial avant d'envoyer l'automatisation à deux autres ressources. Le système envoie de façon exponentielle l'automatisation à davantage de ressources jusqu'à ce que la valeur de simultanéité soit atteinte.

Seuils d'erreurs

Un seuil d'erreurs vous permet de spécifier le nombre d'automatisations autorisées à échouer avant qu' ne Systems Manager cesse d'envoyer l'automatisation à d'autres ressources. Vous pouvez spécifier un nombre absolu d'erreurs, par exemple 10, ou un pourcentage de l'ensemble de la cible, par exemple 10 %.

Par exemple, si vous spécifiez un nombre absolu de 3 erreurs, le système cesse d'exécuter l'automatisation à la réception de la quatrième erreur. Si vous spécifiez 0, le système cesse d'exécuter l'automatisation sur des cibles supplémentaires après le premier résultat d'erreur renvoyé.

Si vous envoyez une automatisation à, par exemple, 50 instances et que vous définissez le seuil d'erreurs sur 10 %, le système cesse d'envoyer la commande à des instances supplémentaires à la réception de la cinquième erreur. Les appels qui exécutent déjà une automatisation lorsqu'un seuil d'erreurs est atteint sont autorisés à se terminer, mais certaines de ces automatisations peuvent également échouer. Si vous devez vous assurer qu'il n'y aura pas plus d'erreurs que le nombre spécifié pour le seuil d'erreurs, définissez la valeur de simultanéité sur 1 afin que les automatisations s'exécutent une par une.

Exécution d'automatisations basées sur des déclencheurs

Cette section inclut des informations sur la façon d'exécuter des automatisations à l'aide d'un déclencheur. Les automatisations peuvent être initiées par plusieurs déclencheurs, tels que Amazon EventBridge des , State Manager des associations ou des fenêtres de maintenance. En utilisant des déclencheurs, vous pouvez exécuter des automatisations à la suite d'un événement spécifique ou de manière planifiée.

Table des matières

- Exécution d'automatisations avec des déclencheurs à l'aide deEventBridge (p. 343)
- Exécution d'automatisations avec des déclencheurs à l'aide deState Manager (p. 349)
- Exécution d'automatisations avec des déclencheurs à l'aide deFenêtres de maintenance (p. 357)

Exécution d'automatisations avec des déclencheurs à l'aide deEventBridge

Vous pouvez démarrer une automatisation en spécifiant un document Automation en tant que cible d'un Amazon EventBridge événement. Vous pouvez démarrer des flux de travail selon un programme ou lorsqu'un événement de système AWS spécifique se produit. Par exemple, supposons que vous créez un document Automation nommé BootStrapInstances qui installe un logiciel sur une instance lorsqu'une instance démarre. Pour spécifier le BootStrapInstances document (et le flux de travail correspondant)

en tant que cible d'un EventBridge événement, vous devez d'abord créer une nouvelle EventBridge règle. (Voici un exemple de règle : nom de service : EC2, type d'événement : notification de changement d'état de l'instance EC2, état(s) spécifique(s) : en cours d'exécution, n'importe quelle instance.) Ensuite, vous utilisez les procédures suivantes pour spécifier le BootStrapInstances document en tant que cible de l'événement à l'aide de la EventBridge console , de l' AWS Command Line Interface (AWS CLI) ou des Outils AWS pour Windows PowerShell . Lors du démarrage d'une nouvelle instance, le système exécute le flux de travail et installe le logiciel.

Pour de plus amples informations sur la création de documents Automation, veuillez consulter Utilisation des documents Automation (p. 446).

Création d'un EventBridge événement qui exécute une automatisation (console)

Utilisez la procédure suivante afin de configurer une automatisation en tant que cible d'un EventBridge événement.

Pour configurer un document Automation en tant que cible d'une règle d'événement EventBridge

1. Ouvrez la console Amazon EventBridge à l'adresse https://console.aws.amazon.com/events/.
2. Dans le volet de navigation, choisissez Rules (Règles), puis Create rule (Créer une règle).

   -ou-

   Si la page Amazon EventBridge d'accueil s'ouvre d'abord, choisissez Create rule (Créer une règle).
3. Saisissez un nom et une description pour la règle.

   Une règle ne peut pas avoir le même nom qu'une autre règle de la même région et sur le même bus d'événement.
4. Pour Définir un modèle, choisissez Modèle d'événement ou Programme. Le modèle d'événement vous permet de créer une règle qui génère des événements pour des actions spécifiques dans AWS les services . Programme vous permet de constituer une règle générant des événements selon un programme que vous spécifiez à l'aide du format cron.
5. Choisissez les options restantes pour la règle que vous souhaitez créer, puis choisissez Add target (Ajouter une cible).

   Pour plus d'informations sur la création EventBridge de règles, consultez Mise en route avec Amazon EventBridge dans le Guide de l'utilisateur Amazon EventBridge .
6. Pour Select event bus (Sélectionner un bus d'événement), choisissez le bus d'événement que vous souhaitez associer à cette règle. Si vous souhaitez que cette règle se déclenche sur les événements correspondants provenant de votre propre compte AWS, sélectionnez AWS default event bus (Bus d'événement par défaut AWS). Lorsqu'un service AWS de votre compte émet un événement, il accède toujours au bus d'événement par défaut de votre compte.
7. Pour Target (Cible), choisissez SSM Automation (Automatisation SSM).
8. Pour Document, choisissez un document Automation à exécuter lorsque votre cible est appelée.
9. Développez la version du document Configure et choisissez une version. $DEFAULT a été explicitement défini en tant que version de document par défaut dans Systems Manager . Vous pouvez choisir une version spécifique ou utiliser la dernière version.
10. Déployez Configurer le(s) paramètre(s) d'automatisation et conservez les valeurs de paramètre par défaut (si disponibles) ou saisissez vos propres valeurs.

    > **Note**
    >
    > Les paramètres requis sont dotés d'une astérisque (*) située à côté du nom du paramètre. Pour créer une cible, vous devez spécifier une valeur pour chaque paramètre requis. Dans le cas contraire, le système crée la règle mais elle ne sera pas exécutée.
11. En bas de la zone Select targets (Sélectionner les cibles), choisissez un rôle pour accorder EventBridge l'autorisation de démarrer le flux de travail Automation avec le document et les

paramètres spécifiés. EventBridge utilise le rôle pour démarrer l'automatisation. Vous pouvez laisser EventBridge créer un nouveau rôle ou utiliser un rôle qui possède déjà les autorisations nécessaires.

12. (Facultatif) Entrez une ou plusieurs balises pour la règle. Pour de plus amples informations, veuillez consulter Balisage de vos ressources Amazon EventBridge dans le Guide de l'utilisateur Amazon EventBridge.
13. Choisissez Create et exécutez l'assistant.

Créer un EventBridge événement qui exécute un document Automation (ligne de commande)

La procédure suivante décrit comment utiliser l' AWS CLI (sous Linux ou Windows) ou Outils AWS pour PowerShell pour créer une règle EventBridge d'événement et configurer un document Automation en tant que cible.

Pour configurer un document Automation en tant que cible d'une règle EventBridge d'événement

1. Installez et configurez l'AWS CLI ou Outils AWS pour PowerShell si vous ne l'avez pas déjà fait.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).
2. Créez une commande pour spécifier une nouvelle règle d'événement EventBridge. Pour vous aider, voici quelques modèles de commandes.

   Déclencheurs basés sur un calendrier

   Linux

   ```
   aws events put-rule \
       --name "rule_name" \
       --schedule-expression "cron_or_rate_expression"
   ```

   Windows

   ```
   aws events put-rule ^
       --name "rule_name" ^
       --schedule-expression "cron_or_rate_expression"
   ```

   PowerShell

   ```
   Write-CWERule `
       -Name "rule_name" `
       -ScheduleExpression "cron_or_rate_expression"
   ```

   L'exemple suivant crée une règle EventBridge d'événement qui se déclenche tous les jours à 9h00 (UTC).

   Linux

   ```
   aws events put-rule \
       --name "DailyAutomationRule" \
       --schedule-expression "cron(0 9 * * ? *)"
   ```

   Windows

   ```
   aws events put-rule ^
       --name "DailyAutomationRule" ^
   ```

```
    --schedule-expression "cron(0 9 * * ? *)"
```

PowerShell

```
Write-CWERule `
    -Name "DailyAutomationRule" `
    -ScheduleExpression "cron(0 9 * * ? *)"
```

Déclencheurs basés sur un événement

Linux

```
aws events put-rule \
    --name "rule_name" \
    --event-pattern "{\"source\":[\"aws.service\"],\"detail-type\":
[\"service_event_detail_type\"]}"
```

Windows

```
aws events put-rule ^
    --name "rule_name" ^
    --event-pattern "{\"source\":[\"aws.service\"],\"detail-type\":
[\"service_event_detail_type\"]}"
```

PowerShell

```
Write-CWERule `
    -Name "rule_name" `
    -EventPattern '{"source":["aws.service"],"detail-type":
["service_event_detail_type"]}'
```

L'exemple suivant crée une règle EventBridge d'événement qui se déclenche lorsqu'une instance EC2 de la région change d'état.

Linux

```
aws events put-rule \
    --name "EC2InstanceStateChanges" \
    --event-pattern "{\"source\":[\"aws.ec2\"],\"detail-type\":[\"EC2 Instance
 State-change Notification\"]}"
```

Windows

```
aws events put-rule ^
    --name "EC2InstanceStateChanges" ^
    --event-pattern "{\"source\":[\"aws.ec2\"],\"detail-type\":[\"EC2 Instance
 State-change Notification\"]}"
```

PowerShell

```
Write-CWERule `
    -Name "EC2InstanceStateChanges" `
    -EventPattern '{"source":["aws.ec2"],"detail-type":["EC2 Instance State-change
 Notification"]}'
```

La commande renvoie des détails concernant la nouvelle règle EventBridge similaires à ce qui suit.

Linux

```
{
    "RuleArn": "arn:aws:events:us-east-1:123456789012:rule/automationrule"
}
```

Windows

```
{
    "RuleArn": "arn:aws:events:us-east-1:123456789012:rule/automationrule"
}
```

PowerShell

```
arn:aws:events:us-east-1:123456789012:rule/EC2InstanceStateChanges
```

3. Créez une commande pour spécifier un document Automation comme cible de la règle d'événement EventBridge que vous avez créée à l'étape 2. Pour vous aider, voici quelques modèles de commandes.

Linux

```
aws events put-targets \
    --rule CW_Event_Rule_Name \
    --targets '{"Arn": "arn:aws:ssm:us-east-1:123456789012:automation-
definition/Automation_Document_Name","Input":"{\"DocumentParameter\":
[\"ParameterValue\"],\"AutomationAssumeRole\":
[\"arn:aws:iam::123456789012:role/AutomationServiceRole\"]}","Id":
 "Target_Id","RoleArn": "arn:aws:iam::123456789012:role/service-
role/CWE_Role_Name_To_Run_Automation"}'
```

Windows

```
aws events put-targets ^
    --rule CW_Event_Rule_Name ^
    --targets '{"Arn": "arn:aws:ssm:us-east-1:123456789012:automation-
definition/Automation_Document_Name","Input":"{\"DocumentParameter\":
[\"ParameterValue\"],\"AutomationAssumeRole\":
[\"arn:aws:iam::123456789012:role/AutomationServiceRole\"]}","Id":
 "Target_Id","RoleArn": "arn:aws:iam::123456789012:role/service-
role/CWE_Role_Name_To_Run_Automation"}'
```

PowerShell

```
$Target = New-Object Amazon.CloudWatchEvents.Model.Target
$Target.Id = "Target_Id"
$Target.Arn = "arn:aws:ssm:us-east-1:123456789012:automation-
definition/Automation_Document_Name"
$Target.RoleArn = "arn:aws:iam::123456789012:role/service-
role/CWE_Role_Name_To_Run_Automation"
$Target.Input = '{"DocumentParameter":["DocumentValue"],"AutomationAssumeRole":
["arn:aws:iam::123456789012:role/AutomationServiceRole"]}'

Write-CWETarget `
    -Rule "CW_Event_Rule_Name" `
```

```
    -Target $Target
```

L'exemple suivant crée une cible EventBridge d'événement qui démarre l'ID d'instance spécifié à l'aide du document `AWS-StartEC2Instance` .

Linux

```
aws events put-targets \
    --rule DailyAutomationRule \
    --targets '{"Arn": "arn:aws:ssm:us-east-1:123456789012:automation-definition/
AWS-StartEC2Instance","Input":"{\"InstanceId\":[\"i-02573cafcfEXAMPLE\"],
\"AutomationAssumeRole\":[\"arn:aws:iam::123456789012:role/AutomationServiceRole
\"]}","Id": "Target1","RoleArn": "arn:aws:iam::123456789012:role/service-role/
AWS_Events_Invoke_Start_Automation_Execution_1213609520"}'
```

Windows

```
aws events put-targets ^
    --rule DailyAutomationRule ^
    --targets '{"Arn": "arn:aws:ssm:us-east-1:123456789012:automation-definition/
AWS-StartEC2Instance","Input":"{\"InstanceId\":[\"i-02573cafcfEXAMPLE\"],
\"AutomationAssumeRole\":[\"arn:aws:iam::123456789012:role/AutomationServiceRole
\"]}","Id": "Target1","RoleArn": "arn:aws:iam::123456789012:role/service-role/
AWS_Events_Invoke_Start_Automation_Execution_1213609520"}'
```

PowerShell

```
$Target = New-Object Amazon.CloudWatchEvents.Model.Target
$Target.Id = "Target1"
$Target.Arn = "arn:aws:ssm:us-east-1:123456789012:automation-definition/AWS-
StartEC2Instance"
$Target.RoleArn = "arn:aws:iam::123456789012:role/service-role/
AWS_Events_Invoke_Start_Automation_Execution_1213609520"
$Target.Input = '{"InstanceId":["i-02573cafcfEXAMPLE"],"AutomationAssumeRole":
["arn:aws:iam::123456789012:role/AutomationServiceRole"]}'

Write-CWETarget `
    -Rule "DailyAutomationRule" `
    -Target $Target
```

Le système renvoie des informations similaires à celles qui suivent.

Linux

```
{
    "FailedEntries": [],
    "FailedEntryCount": 0
}
```

Windows

```
{
    "FailedEntries": [],
    "FailedEntryCount": 0
}
```

PowerShell

Il n'y a pas de sortie si la commande réussit pour PowerShell .

Exécution d'automatisations avec des déclencheurs à l'aide deState Manager

Vous pouvez démarrer une automatisation en créant une State Manager association avec un document Automation. En créant une association State Manager avec un document Automation, vous pouvez cibler différents types de ressources AWS. Par exemple, vous pouvez créer des associations qui appliquent un état souhaité sur une ressource AWS, y compris les éléments suivants :

- Attacher un rôle Systems Manager aux instances EC2 pour en faire des instances gérées.
- Appliquer les règles d'entrée et de sortie souhaitées pour un groupe de sécurité.
- Créer ou supprimer des sauvegardes Amazon DynamoDB (DynamoDB).
- Créer ou supprimer des instantanés Amazon Elastic Block Store (Amazon EBS).
- Désactiver les autorisations de lecture et d'écriture sur les compartiments Amazon Simple Storage Service (Amazon S3).
- Démarrer, redémarrer ou arrêter les instances gérées et les instances Amazon Relational Database Service (Amazon RDS).
- Correctif pour Windows et LinuxAMIs.

Utilisez les procédures suivantes pour créer une State Manager association qui exécute une automatisation à l'aide de la AWS Systems Manager console , AWS Command Line Interface de l' (AWS CLI) ou d' Outils AWS pour Windows PowerShell .

Avant de commencer

Tenez compte des détails importants suivants avant d'exécuter l'automatisation à l'aide d' State Manager .

- Avant de pouvoir créer une association qui exécute un document Automation, vérifiez que vous avez configuré les autorisations pour Systems Manager Automation. Pour plus d'informations, consultez la section Configuration d'Automation (p. 308).
- Les associations State Manager qui exécutent des documents Automation contribuent au nombre maximal d'instances d'Automation exécutées simultanément dans votre compte AWS. Vous pouvez avoir un maximum de 100 automatisations simultanées en cours d'exécution. Pour plus d'informations, consultez la section Quotas de service Systems Manager dans le Référence générale d'Amazon Web Services..
- Systems Manager crée automatiquement un rôle lié à un service afin que State Manager ait l'autorisation d'appeler des actions d'API de Systems Manager Automation. Si vous le souhaitez, vous pouvez créer le rôle lié à un service vous-même en exécutant la commande suivante à partir de l'AWS CLI ou des Outils AWS pour PowerShell.

  Linux

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

  Windows

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

PowerShell

```
New-IAMServiceLinkedRole `
    -AWSServiceName ssm.amazonaws.com
```

Pour plus d'informations sur les rôles liés à un service, consultez Utilisation des rôles liés à un service pour Systems Manager (p. 1312).

Création d'une association qui exécute une automatisation (console)

La procédure suivante décrit comment utiliser la Systems Manager console pour créer une State Manager association qui exécute une automatisation.

Pour créer une State Manager association qui exécute une automatisation

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez State Manager, puis Créer une association.
3. Dans le champ Nom, spécifiez un nom. Cette action est facultative, mais recommandée.
4. Dans la liste Document, choisissez un document. Utilisez la barre de recherche pour filtrer sur le type de document : Égal : Documents d'automatisation. Pour afficher plus de documents Automation, utilisez les nombres à droite de la barre de recherche.

   Note

   Pour afficher des informations sur un document, sélectionnez son nom.
5. Choisissez Simple execution (Exécution simple) pour exécuter l'instance d'Automation sur une ou plusieurs cibles en spécifiant les ID de ressource pour ces cibles. Choisissez Rate Control (Contrôle du débit) pour exécuter l'instance d'Automation sur une flotte de ressources AWS en spécifiant une option de ciblage telles que les balises ou Groupes de ressources AWS. Vous pouvez également contrôler l'exécution de l'instance d'Automation à travers vos ressources en spécifiant la simultanéité et les seuils d'erreur.

   Si vous avez choisi Rate Control (Contrôle du débit), la section Targets (Cibles) apparaît.
6. Dans la section Targets (Cibles), choisissez une méthode pour cibler des ressources.

   a. (Facultatif) Dans la liste Parameter (Paramètre), sélectionnez un paramètre. Les éléments de la liste Parameter (Paramètre) sont déterminés par les paramètres du document d'Automation que vous avez sélectionnés au début de cette procédure. En choisissant un paramètre, vous définissez le type de ressource sur lequel l'automatisation s'exécute.
   b. (Facultatif) Dans la liste Targets (Cibles), choisissez une méthode pour cibler les ressources.

      - Groupe de ressources : Choisissez le nom du groupe dans la liste Groupe de ressources. Pour de plus amples informations sur la cible Groupes de ressources AWS dans les documents Automation, veuillez consulter Ciblage des Groupes de ressources AWS (p. 339).
      - Balises : Entrez la clé de balise et (éventuellement) la valeur de balise dans les champs fournis. Choisissez Add (Ajouter). Pour de plus amples informations sur le ciblage des balises dans les documents Automation, veuillez consulter Ciblage des balises (p. 337).
      - Valeurs des paramètres : Saisissez les valeurs dans la section Paramètres d'entrée. Si vous spécifiez plusieurs valeurs, Systems Manager exécute une automatisation enfant sur chaque valeur spécifiée.

        Par exemple, supposons que votre document Automation inclut un InstanceID paramètre . Si vous ciblez les valeurs du InstanceID paramètre lorsque vous exécutez l'automatisation, Systems Manager exécute une automatisation enfant pour chaque valeur d'ID d'instance

spécifiée. L'automatisation parent est terminée lorsque l'automatisation termine l'exécution de chaque instance spécifiée ou si l'automatisation échoue. Vous pouvez cibler un maximum de 50 valeurs de paramètre. Pour de plus amples informations sur le ciblage des valeurs de paramètres dans les documents Automation, veuillez consulter Ciblage des paramètres de valeur (p. 340).

7. Dans la section Paramètres d'entrée, spécifiez les paramètres d'entrée obligatoires.

   Si vous choisissez de cibler les ressources en utilisant des balises ou un groupe de ressources, certaines options de la section Paramètres d'entrée n'ont pas besoin d'être définies. Par exemple, si vous avez choisi le document AWS-RestartEC2Instance et que vous avez choisi de cibler des instances à l'aide de balises, vous n'avez pas besoin de spécifier ou de choisir une instance IDs dans la section Paramètres d'entrée. L'automatisation localise les instances à redémarrer en utilisant les balises que vous avez spécifiées.

   > **Important**
   >
   > Vous devez spécifier un ARN de rôle dans le AutomationAssumeRole champ. State Manager utilise le rôle assume pour appeler AWS les services spécifiés dans le document Automation et exécuter des associations Automation en votre nom. Pour plus d'informations, consultez la section Exécution d'une automatisation à l'aide d'un rôle IAM de service (p. 368).

8. Dans la section Spécifier le programme, choisissez On Schedule (Activé) si vous voulez exécuter l'association à intervalles réguliers. Si vous choisissez cette option, utilisez les options fournies pour créer le calendrier à l'aide des expressions Cron ou Rate. Pour de plus amples informations sur les expressions de type Cron et Rate pour State Manager, veuillez consulter Expressions cron et rate pour les associations (p. 1397).

   > **Note**
   >
   > Les expressions Rate sont le mécanisme de planification préféré pour les associations State Manager qui exécutent des documents Automation. Les expressions Rate permettent plus de flexibilité pour exécuter des associations si vous atteignez le nombre maximal d'automatisations exécutées simultanément. Avec une planification Rate, Systems Manager peut réessayer l'automatisation peu de temps après avoir reçu une notification indiquant que les automatisations simultanées ont atteint leur maximum et ont été limitées.

   Choisissez No schedule (Aucun programme) si vous souhaitez exécuter l'association à un moment donné.

9. (Facultatif) Dans la section Contrôle du débit, choisissez des options de simultanéité et de seuil d'erreurs pour contrôler le déploiement d'Automation dans vos AWS ressources.

   a. Dans la section Simultanéité, choisissez une option :

      - Choisissez des cibles pour entrer un nombre absolu de cibles pouvant exécuter l'automatisation simultanément.
      - Choisissez un pourcentage pour entrer un pourcentage de l'ensemble de cibles pouvant exécuter l'automatisation simultanément.

   b. Dans la section Error threshold (Seuil d'erreurs), choisissez une option :

      - Choisissez erreurs pour indiquer un nombre absolu d'erreurs autorisées avant qu'Automation ne cesse d'envoyer le flux de travail à d'autres ressources.
      - Choisissez pourcentage pour indiquer un pourcentage d'erreurs autorisées avant qu'Automation ne cesse d'envoyer le flux de travail à d'autres ressources.

   Pour de plus amples informations sur l'utilisation des cibles et du contrôle du débit avec Automation, veuillez consulter Exécution d'automatisations qui utilisent des cibles et des contrôles du débit (p. 328).

10. Choisissez Create Association (Créer une association).

    Important

    Lorsque vous créez une association, l'association s'exécute immédiatement les cibles spécifiées. L'association s'exécute ensuite en fonction de l'expression cron ou rate que vous avez choisie. Si vous choisissez No schedule (Aucun programme), l'association n'est pas exécutée à nouveau.

Création d'une association qui exécute une automatisation (ligne de commande)

La procédure suivante décrit comment utiliser l' AWS CLI (sous Linux ou Windows) ou Outils AWS pour PowerShell pour créer une State Manager association qui exécute une automatisation.

Pour créer une association qui exécute une automatisation

1. Installez et configurez l'AWS CLI ou Outils AWS pour PowerShell si vous ne l'avez pas déjà fait.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).
2. Exécutez la commande suivante pour afficher une liste de documents.

   Linux

   ```
   aws ssm list-documents
   ```

   Windows

   ```
   aws ssm list-documents
   ```

   PowerShell

   ```
   Get-SSMDocumentList
   ```

   Notez le nom du document Automation que vous souhaitez utiliser pour l'association.
3. Exécutez la commande suivante pour afficher des informations détaillées relatives au document Automation.

   Linux

   ```
   aws ssm describe-document \
       --name document_name
   ```

   Notez un nom de paramètre (par exemple, `InstanceId`) que vous souhaitez utiliser pour l'option `--automation-target-parameter-name`. Ce paramètre détermine le type de ressource sur lequel l'automatisation s'exécute.

   Windows

   ```
   aws ssm describe-document ^
       --name document_name
   ```

   Notez un nom de paramètre (par exemple, `InstanceId`) que vous souhaitez utiliser pour l'option `--automation-target-parameter-name`. Ce paramètre détermine le type de ressource sur lequel l'automatisation s'exécute.

PowerShell

```
Get-SSMDocumentDescription `
    -Name document_name
```

Notez un nom de paramètre (par exemple, `InstanceId`) que vous souhaitez utiliser pour l'option `AutomationTargetParameterName`. Ce paramètre détermine le type de ressource sur lequel l'automatisation s'exécute.

4. Créez une commande qui exécute une automatisation à l'aide d'une State Manager association. Pour vous aider, voici quelques modèles de commandes.

   Ciblage à l'aide de balises

   Linux

```
aws ssm create-association \
    --association-name AssociationName \
    --targets Key=tag:TagKey,Values=TagValue \
    --name AutomationDocumentName \
    --parameters AutomationAssumeRole=arn:aws:iam::123456789012:role/aws-service-
role/ssm.amazonaws.com/AWSServiceRoleForAmazonSSM,(Additional parameters, if any) \
    --automation-target-parameter-name (parameter to target) \
    --schedule "cron_or_rate_expression"
```

   > **Note**
   >
   > Si vous créez une association à l'aide des AWS CLI, utilisez le paramètre `--targets` pour cibler les instances pour l'association. N'utilisez pas le paramètre `--instance-id`. Le paramètre `--instance-id` est un paramètre hérité.

   Windows

```
aws ssm create-association ^
    --association-name AssociationName ^
    --targets Key=tag:TagKey,Values=TagValue ^
    --name AutomationDocumentName ^
    --parameters AutomationAssumeRole=arn:aws:iam::123456789012:role/aws-service-
role/ssm.amazonaws.com/AWSServiceRoleForAmazonSSM,(Additional parameters, if any) ^
    --automation-target-parameter-name (parameter to target) ^
    --schedule "cron_or_rate_expression"
```

   > **Note**
   >
   > Si vous créez une association à l'aide des AWS CLI, utilisez le paramètre `--targets` pour cibler les instances pour l'association. N'utilisez pas le paramètre `--instance-id`. Le paramètre `--instance-id` est un paramètre hérité.

   PowerShell

```
$Targets = New-Object Amazon.SimpleSystemsManagement.Model.Target
$Targets.Key = "tag:TagKey"
$Targets.Values = "TagValue"

New-SSMAssociation `
    -AssociationName "AssociationName" `
    -Target $Targets `
    -Name "AutomationDocumentName" `
    -Parameters @{
    "AutomationAssumeRole"="arn:aws:iam::123456789012:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM;
```

```
    (Additional parameters, if any)" } `
    -AutomationTargetParameterName "parameter_to_target" `
    -ScheduleExpression "cron_or_rate_expression"
```

> **Note**
>
> Si vous créez une association à l'aide des Outils AWS pour PowerShell, utilisez le paramètre `Target` pour cibler les instances pour l'association. N'utilisez pas le paramètre `InstanceId`. Le paramètre `InstanceId` est un paramètre hérité.

Ciblage à l'aide des valeurs de paramètre

Linux

```
aws ssm create-association \
    --association-name AssociationName \
    --targets Key=ParameterValues,Values=value_1,value_2,value_3 \
    --name AutomationDocumentName \
    --parameters AutomationAssumeRole=arn:aws:iam::123456789012:role/aws-service-
role/ssm.amazonaws.com/AWSServiceRoleForAmazonSSM,(Additional parameters, if any) \
    --automation-target-parameter-name (parameter to target) \
    --schedule "cron_or_rate_expression"
```

Windows

```
aws ssm create-association ^
    --association-name AssociationName ^
    --targets Key=ParameterValues,Values=value_1,value_2,value_3 ^
    --name AutomationDocumentName ^
    --parameters AutomationAssumeRole=arn:aws:iam::123456789012:role/aws-service-
role/ssm.amazonaws.com/AWSServiceRoleForAmazonSSM,(Additional parameters, if any) ^
    --automation-target-parameter-name (parameter to target) ^
    --schedule "cron_or_rate_expression"
```

PowerShell

```
$Targets = New-Object Amazon.SimpleSystemsManagement.Model.Target
$Targets.Key = "ParameterValues"
$Targets.Values = "value_1","value_2","value_3"

New-SSMAssociation `
    -AssociationName "AssociationName" `
    -Target $Targets `
    -Name "AutomationDocumentName" `
    -Parameters @{
    "AutomationAssumeRole"="arn:aws:iam::123456789012:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM;
    (Additional parameters, if any)"} `
    -AutomationTargetParameterName "parameter_to_target" `
    -ScheduleExpression "cron_or_rate_expression"
```

Ciblage à l'aide de Groupes de ressources AWS

Linux

```
aws ssm create-association \
    --association-name AssociationName \
    --targets Key=ResourceGroup,Values=Resource_Group_name \
```

```
    --name AutomationDocumentName \
    --parameters AutomationAssumeRole=arn:aws:iam::123456789012:role/aws-service-
role/ssm.amazonaws.com/AWSServiceRoleForAmazonSSM,(Additional parameters, if any) \
    --automation-target-parameter-name (parameter to target) \
    --schedule "cron_or_rate_expression"
```

Windows

```
aws ssm create-association ^
    --association-name AssociationName ^
    --targets Key=ResourceGroup,Values=Resource_Group_name ^
    --name AutomationDocumentName ^
    --parameters AutomationAssumeRole=arn:aws:iam::123456789012:role/aws-service-
role/ssm.amazonaws.com/AWSServiceRoleForAmazonSSM,(Additional parameters, if any) ^
    --automation-target-parameter-name (parameter to target) ^
    --schedule "cron_or_rate_expression"
```

PowerShell

```
$Targets = New-Object Amazon.SimpleSystemsManagement.Model.Target
$Targets.Key = "ResourceGroup"
$Targets.Values = "Resource_Group_Name"

New-SSMAssociation `
    -AssociationName "AssociationName" `
    -Target $Targets `
    -Name "AutomationDocumentName" `
    -Parameters @{
    "AutomationAssumeRole"="arn:aws:iam::123456789012:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM;
    (Additional parameters, if any)"} `
    -AutomationTargetParameterName "parameter_to_target" `
    -ScheduleExpression "cron_or_rate_expression"
```

La commande renvoie des détails pour la nouvelle association similaires à ce qui suit.

Linux

```
{
    "AssociationDescription": {
        "ScheduleExpression": "cron(0 7 ? * MON *)",
        "Name": "AWS-StartEC2Instance",
        "Parameters": {
            "AutomationAssumeRole": [
                "arn:aws:iam::123456789012:role/aws-service-role/ssm.amazonaws.com/
AWSServiceRoleForAmazonSSM"
            ]
        },
        "Overview": {
            "Status": "Pending",
            "DetailedStatus": "Creating"
        },
        "AssociationId": "1450b4b7-bea2-4e4b-b340-01234EXAMPLE",
        "DocumentVersion": "$DEFAULT",
        "AutomationTargetParameterName": "InstanceId",
        "LastUpdateAssociationDate": 1564686638.498,
        "Date": 1564686638.498,
        "AssociationVersion": "1",
        "AssociationName": "CLI",
        "Targets": [
            {
```

```
                "Values": [
                    "DEV"
                ],
                "Key": "tag:ENV"
            }
        ]
    }
}
```

Windows

```
{
    "AssociationDescription": {
        "ScheduleExpression": "cron(0 7 ? * MON *)",
        "Name": "AWS-StartEC2Instance",
        "Parameters": {
            "AutomationAssumeRole": [
                "arn:aws:iam::123456789012:role/aws-service-role/ssm.amazonaws.com/
AWSServiceRoleForAmazonSSM"
            ]
        },
        "Overview": {
            "Status": "Pending",
            "DetailedStatus": "Creating"
        },
        "AssociationId": "1450b4b7-bea2-4e4b-b340-01234EXAMPLE",
        "DocumentVersion": "$DEFAULT",
        "AutomationTargetParameterName": "InstanceId",
        "LastUpdateAssociationDate": 1564686638.498,
        "Date": 1564686638.498,
        "AssociationVersion": "1",
        "AssociationName": "CLI",
        "Targets": [
            {
                "Values": [
                    "DEV"
                ],
                "Key": "tag:ENV"
            }
        ]
    }
}
```

PowerShell

```
Name                  : AWS-StartEC2Instance
InstanceId            :
Date                  : 8/1/2019 7:31:38 PM
Status.Name           :
Status.Date           :
Status.Message        :
Status.AdditionalInfo :
```

Note

Si vous utilisez des balises pour créer une association sur une ou plusieurs instances cibles, puis que vous supprimez les balises d'une instance, cette instance n'exécute plus l'association. L'instance est dissociée du document State Manager.

Dépannage des State Managerautomatisations

Systems ManagerAutomation impose une limite de 100 automatisations simultanées et 1 000 automatisations mises en file d'attente par compte et par région. Si une State Manager association qui exécute un document Automation affiche le statut Failed et le statut détaillé AutomationExecutionLimitExceeded , votre exécution peut avoir atteint la limite. Par conséquent, Systems Manager limite les exécutions. Pour résoudre ce problème, procédez comme suit:

- Utilisez un taux différent ou une expression cron différente pour votre association. Par exemple, si l'exécution de l'association est prévue toutes les 30 minutes, alors modifiez l'expression de telle sorte qu'elle s'exécute toutes les heures ou toutes les deux heures.
- Supprimez les automatisations existantes qui ont le statut En attente. En supprimant ces automatisations, vous effacez la file d'attente actuelle.

Exécution d'automatisations avec des déclencheurs à l'aide deFenêtres de maintenance

Vous pouvez démarrer une automatisation en configurant un document Automation en tant que tâche enregistrée pour une fenêtre de maintenance. En enregistrant le document Automation en tant que tâche enregistrée, la fenêtre de maintenance exécute l'automatisation au cours de la période de maintenance planifiée.

Par exemple, supposons que vous créiez un document Automation nommé CreateAMI qui crée une Amazon Machine Image (AMI) des instances enregistrées en tant que cibles dans la fenêtre de maintenance. Pour spécifier le CreateAMI document (et le flux de travail correspondant) en tant que tâche enregistrée d'une fenêtre de maintenance, vous devez d'abord créer une fenêtre de maintenance et enregistrer des cibles. Ensuite, vous utilisez la procédure suivante pour spécifier le CreateAMI document en tant que tâche enregistrée dans la fenêtre de maintenance. Lorsque la fenêtre de maintenance démarre au cours de la période planifiée, le système exécute le flux de travail Automation et crée une AMI de cibles enregistrées.

Pour de plus amples informations sur la création de documents Automation, veuillez consulter Utilisation des documents Automation (p. 446).

Utilisez les procédures suivantes afin de configurer une automatisation en tant que tâche enregistrée pour une fenêtre de maintenance à l'aide de la AWS Systems Manager console , de l' AWS Command Line Interface (AWS CLI) ou des Outils AWS pour Windows PowerShell .

Enregistrement d'une tâche d'automatisation dans une fenêtre de maintenance (console)

La procédure suivante décrit comment utiliser la Systems Manager console pour configurer une automatisation en tant que tâche enregistrée pour une fenêtre de maintenance.

Avant de commencer

Avant d'exécuter la procédure suivante, vous devez créer une fenêtre de maintenance et enregistrer au moins une cible. Pour de plus amples informations, veuillez consulter les procédures suivantes.

- Créer une fenêtre de maintenance (console) (p. 707).
- Affecter des cibles à une fenêtre de maintenance (console) (p. 709)

Pour configurer une automatisation en tant que tâche enregistrée pour une fenêtre de maintenance

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.

2. Dans le volet de navigation de gauche, choisissez Fenêtres de maintenance , puis choisissez la fenêtre de maintenance avec laquelle vous souhaitez enregistrer une tâche d'automatisation.
3. Choisissez Actions. Ensuite, choisissez Register Automation task (Enregistrer la tâche d'automatisation) pour exécuter l'automatisation de votre choix sur les cibles à l'aide d'un document d'automatisation.
4. Pour Name (Nom), entrez le nom de la tâche.
5. Pour Description, entrez une description.
6. Pour Document, choisissez le document Automation qui définit les tâches à exécuter.
7. Pour Version du document, choisissez la version de document à utiliser.
8. Pour la priorité de tâche, spécifiez une priorité pour cette tâche. 1 est la priorité la plus élevée. Les tâches d'une fenêtre de maintenance sont planifiées par ordre de priorité. Les tâches qui ont la même priorité sont planifiées en parallèle.
9. Dans la section Cibles, identifiez les cibles sur lesquelles vous voulez exécuter ce flux de travail Automation en spécifiant des balises ou en sélectionnant des instances manuellement.

   Important

   Si vous choisissez un document Automation qui ne cible pas les instances gérées, vous devez sélectionner au moins une cible de fenêtre de maintenance. Dans ce cas, nous vous recommandons d'enregistrer une cible pour une paire clé-valeur de balise qui n'est pas utilisée par vos instances gérées.
   Par exemple, si vous choisissez le document Automation `AWS-CopySnapshot`, le flux de travail Automation des instantanés cible des instantanés Amazon Elastic Block Store (EBS) au lieu des instances gérées. Dans ce cas, vous pouvez enregistrer une cible sur votre fenêtre de maintenance, qui cible une paire valeur-clé de balise qui n'est pas utilisée par vos instances gérées, telle que key=MaintenanceWindow et value=Snapshot.
10. (Facultatif) Pour Contrôle du débit :
    - Pour Simultanéité, spécifiez un nombre ou un pourcentage de cibles sur lesquelles exécuter l'automatisation en même temps.

      Note

      Si vous avez sélectionné des cibles en choisissant des paires clé-valeur de balises, et que vous n'êtes pas sûr de la quantité de cibles utilisant les balises sélectionnées, limitez alors le nombre de flux de travail Automation qui peuvent s'exécuter en même temps en spécifiant un pourcentage.
      Lorsque la fenêtre de maintenance s'exécute, une nouvelle automatisation est lancée par cible. Il existe une limite de 100 automatisations simultanées par AWS compte. Si vous spécifiez un taux de simultanéité supérieur à 100, les automatisations simultanées supérieures à 100 sont automatiquement ajoutées à la file d'attente d'automatisation. Pour plus d'informations, consultez la section Quotas de service Systems Manager dans le Référence générale d'Amazon Web Services..
    - Pour Error threshold (Seuil d'erreur), spécifiez quand arrêter l'exécution de l'automatisation sur d'autres cibles après son échec sur un nombre ou un pourcentage de cibles. Par exemple, si vous spécifiez trois erreurs, Systems Manager arrête l'exécution des automatisations à la réception de la quatrième erreur. Les cibles traitant toujours l'automatisation peuvent également envoyer des erreurs.
11. Dans la zone IAM service role (Rôle de service), choisissez l'une des options suivantes pour autoriser Systems Manager à démarrer l'automatisation :
    - Créer et utiliser un rôle lié à un service pour Systems Manager

      Les rôles liés à un service offrent une manière sécurisée d'accorder des autorisations aux services AWS, car seul le service lié peut assumer un rôle lié à un service. De plus, AWS définit

automatiquement les autorisations des rôles liés à un service, en fonction des actions que le service lié effectue en votre nom.

> **Note**
>
> Si un rôle lié à un service a déjà été créé pour votre compte, choisissez Use the service-linked role for Systems Manager (Utiliser le rôle lié à un service pour Systems Manager).

- Use a custom service role (Utiliser un rôle de service personnalisé)

  Vous pouvez créer un rôle de service personnalisé pour les tâches de fenêtre de maintenance si vous souhaitez utiliser des autorisations plus rigoureuses que celles fournies par le rôle lié à un service.

  Si vous avez besoin de créer un rôle de service personnalisé, veuillez consulter l'une des rubriques suivantes :

  - Contrôle de l'accès aux fenêtres de maintenance (console) (p. 692)
  - Contrôler l'accès aux fenêtres de maintenance (AWS CLI) (p. 697)
  - Contrôler l'accès aux fenêtres de maintenance (Outils pour Windows PowerShell) (p. 703)

Pour choisir plus facilement d'utiliser un rôle de service personnalisé ou le rôle lié à un service Systems Manager avec une tâche de fenêtre de maintenance, consultez Dois-je utiliser un rôle lié à un service ou un rôle de service personnalisé pour exécuter des tâches de fenêtre de maintenance ? (p. 691).

12. Dans la section Input Parameters (Paramètres d'entrée), spécifiez les paramètres du document. Pour les documents Automation, le système fournit automatiquement certaines des valeurs. Vous pouvez conserver ou remplacer ces valeurs.

    > **Important**
    >
    > Pour les documents Automation, vous pouvez éventuellement spécifier un rôle responsable Automation. Si vous ne spécifiez pas de rôle pour ce paramètre, Automation endosse le rôle de service de fenêtre de maintenance que vous avez choisi à l'étape 11. À ce titre, vous devez veiller à ce que le rôle de service de fenêtre de maintenance que vous avez choisi possède les autorisations appropriées AWS Identity and Access Management (IAM) pour effectuer les actions définies dans le document Automation.
    > Par exemple, le rôle lié à un service pour Systems Manager n'a pas l'IAMautorisation `ec2:CreateSnapshot` , qui est requise pour exécuter le document Automation`AWS-CopySnapshot`. Dans ce scénario, vous devez utiliser un rôle de service de fenêtre de maintenance personnalisé ou spécifier un rôle responsable Automation qui dispose `ec2:CreateSnapshot` des autorisations. Pour plus d'informations, consultez la section Configuration d'Automation (p. 308).

13. Choisissez Enregistrer la tâche d'automatisation.

Enregistrement d'une tâche d'automatisation dans une fenêtre de maintenance (ligne de commande)

La procédure suivante décrit comment utiliser l' AWS CLI (sous Linux ou Windows) ou Outils AWS pour PowerShell pour configurer une automatisation en tant que tâche enregistrée pour une fenêtre de maintenance.

Avant de commencer

Avant d'exécuter la procédure suivante, vous devez créer une fenêtre de maintenance et enregistrer au moins une cible. Pour de plus amples informations, veuillez consulter les procédures suivantes.

- Étape 1 : Créer la fenêtre de maintenance (AWS CLI) (p. 717).

- Étape 2 : Enregistrer une instance cible avec la fenêtre de maintenance (AWS CLI) (p. 719)

Pour configurer une automatisation en tant que tâche enregistrée pour une fenêtre de maintenance

1. Installez et configurez l'AWS CLI ou Outils AWS pour PowerShell si vous ne l'avez pas déjà fait.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Créez une commande pour configurer une automatisation en tant que tâche enregistrée pour une fenêtre de maintenance. Pour vous aider, voici quelques modèles de commandes.

   Linux

```
aws ssm register-task-with-maintenance-window \
    --window-id window_id \
    --name task_name \
    --task-arn document_name \
    --targets Key=targets,Values=value_1,value_2,value_3 \
    --service-role-arn service_role_arn \
    --task-type AUTOMATION \
    --task-invocation-parameters task_parameters_if_any \
    --priority task_priority \
    --max-concurrency a_number_of_instances_or_a_percentage_of_target_set \
    --max-errors a_number_of_errors_or_a_percentage_of_target_set
```

   Note

   Si vous configurez une automatisation en tant que tâche enregistrée à l'aide AWS CLI de l' , utilisez le `--Task-Invocation-Parameters` paramètre pour spécifier les paramètres à transmettre à une tâche lorsqu'elle s'exécute. N'utilisez pas le paramètre `--Task-Parameters`. Le paramètre `--Task-Parameters` est un paramètre hérité.

   Windows

```
aws ssm register-task-with-maintenance-window ^
    --window-id window_id ^
    --name task_name ^
    --task-arn document_name ^
    --targets Key=targets,Values=value_1,value_2,value_3 ^
    --service-role-arn service_role_arn ^
    --task-type AUTOMATION ^
    --task-invocation-parameters task_parameters_if_any ^
    --priority task_priority ^
    --max-concurrency a_number_of_instances_or_a_percentage_of_target_set ^
    --max-errors a_number_of_errors_or_a_percentage_of_target_set
```

   Note

   Si vous configurez une automatisation en tant que tâche enregistrée à l'aide AWS CLI de l' , utilisez le `--Task-Invocation-Parameters` paramètre pour spécifier les paramètres à transmettre à une tâche lorsqu'elle s'exécute. N'utilisez pas le paramètre `--Task-Parameters`. Le paramètre `--Task-Parameters` est un paramètre hérité.

   PowerShell

```
Register-SSMTaskWithMaintenanceWindow `
    -WindowId window_id `
    -Name "task_name" `
    -TaskArn "document_name" `
```

```
    -Target @{ Key="targets";Values="value_1" } `
    -ServiceRoleArn "service_role_arn" `
    -TaskType "AUTOMATION" `
    -Automation_Parameter
 @{ "task_parameters_1"="task_parameter_1_value";"task_parameters_2"="task_parameter_2_value" }
 `
    -Priority task_priority `
    -MaxConcurrency a_number_of_instances_or_a_percentage_of_target_set `
    -MaxError a_number_of_errors_or_a_percentage_of_target_set
```

Note

Si vous configurez une automatisation en tant que tâche enregistrée à l'aide Outils AWS pour PowerShell de l' , utilisez le `-Automation_Parameter` paramètre pour spécifier les paramètres à transmettre à une tâche lorsque celle-ci s'exécute. N'utilisez pas le paramètre `-TaskParameters`. Le paramètre `-TaskParameters` est un paramètre hérité.

L'exemple suivant configure une automatisation en tant que tâche enregistrée pour une fenêtre de maintenance avec la priorité 1. L'automatisation utilise le `AWS-StartEC2Instance` document et le rôle de responsable Automation spécifié pour démarrer les instances EC2 enregistrées en tant que cibles dans la fenêtre de maintenance. La fenêtre de maintenance exécute l'automatisation simultanément sur 5 instances maximum à un moment donné. En outre, la tâche enregistrée cesse d'être exécutée sur des instances supplémentaires pour un intervalle d'exécution particulier si le nombre d'erreurs dépasse 1.

Linux

```
aws ssm register-task-with-maintenance-window \
    --window-id mw-0c50858d01EXAMPLE \
    --name StartEC2Instances \
    --task-arn AWS-StartEC2Instance \
    --targets Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE \
    --service-role-arn arn:aws:iam::123456789012:role/MaintenanceWindowRole \
    --task-type AUTOMATION \
    --task-invocation-parameters "{\"Automation\":{\"Parameters\":{\"InstanceId\":
[\"{{TARGET_ID}}\"],\"AutomationAssumeRole\":[\"arn:aws:iam::123456789012:role/
AutomationAssumeRole\"]}}}" \
    --priority 1 \
    --max-concurrency 5 \
    --max-errors 1
```

Windows

```
aws ssm register-task-with-maintenance-window ^
    --window-id mw-0c50858d01EXAMPLE ^
    --name StartEC2Instances ^
    --task-arn AWS-StartEC2Instance ^
    --targets Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE ^
    --service-role-arn arn:aws:iam::123456789012:role/MaintenanceWindowRole ^
    --task-type AUTOMATION ^
    --task-invocation-parameters "{\"Automation\":{\"Parameters\":{\"InstanceId\":
[\"{{TARGET_ID}}\"],\"AutomationAssumeRole\":[\"arn:aws:iam::123456789012:role/
AutomationAssumeRole\"]}}}" ^
    --priority 1 ^
    --max-concurrency 5 ^
    --max-errors 1
```

PowerShell

```
Register-SSMTaskWithMaintenanceWindow `
    -WindowId mw-0c50858d01EXAMPLE `
    -Name "StartEC2" `
    -TaskArn "AWS-StartEC2Instance" `
    -Target
 @{ Key="WindowTargetIds";Values="e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE" } `
    -ServiceRoleArn "arn:aws:iam::123456789012:role/MaintenanceWindowRole" `
    -TaskType "AUTOMATION" `
    -Automation_Parameter
 @{ "InstanceId"="{{TARGET_ID}}";"AutomationAssumeRole"="arn:aws:iam::123456789012:role/
AutomationAssumeRole" } `
    -Priority 1 `
    -MaxConcurrency 5 `
    -MaxError 1
```

La commande renvoie des détails similaires à ce qui suit pour la nouvelle tâche enregistrée.

Linux

```
{
    "WindowTaskId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE"
}
```

Windows

```
{
    "WindowTaskId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE"
}
```

PowerShell

```
4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE
```

3. Pour afficher la tâche enregistrée, exécutez la commande suivante.

Linux

```
aws ssm describe-maintenance-window-tasks \
    --window-id mw-0c50858d01EXAMPLE
```

Windows

```
aws ssm describe-maintenance-window-tasks ^
    --window-id mw-0c50858d01EXAMPLE
```

PowerShell

```
Get-SSMMaintenanceWindowTaskList `
    -WindowId mw-0c50858d01EXAMPLE
```

Le système renvoie des informations similaires à celles qui suivent.

Linux

```
{
    "Tasks": [
        {
            "ServiceRoleArn": "arn:aws:iam::123456789012:role/
MaintenanceWindowRole",
            "MaxErrors": "1",
            "TaskArn": "AWS-StartEC2Instance",
            "MaxConcurrency": "5",
            "WindowTaskId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE",
            "TaskParameters": {},
            "Priority": 0,
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Type": "AUTOMATION",
            "Targets": [
                {
                    "Values": [
                        "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
                    ],
                    "Key": "WindowTargetIds"
                }
            ],
            "Name": "StartEC2"
        }
    ]
}
```

Windows

```
{
    "Tasks": [
        {
            "ServiceRoleArn": "arn:aws:iam::123456789012:role/
MaintenanceWindowRole",
            "MaxErrors": "1",
            "TaskArn": "AWS-StartEC2Instance",
            "MaxConcurrency": "5",
            "WindowTaskId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE",
            "TaskParameters": {},
            "Priority": 0,
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Type": "AUTOMATION",
            "Targets": [
                {
                    "Values": [
                        "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
                    ],
                    "Key": "WindowTargetIds"
                }
            ],
            "Name": "StartEC2"
        }
    ]
}
```

PowerShell

```
Description    :
LoggingInfo    :
MaxConcurrency : 5
MaxErrors      : 1
```

```
Name            : StartEC2
Priority        : 1
ServiceRoleArn  : arn:aws:iam::123456789012:role/MaintenanceWindowRole
Targets         : {WindowTargetIds}
TaskArn         : AWS-StartEC2Instance
TaskParameters  : {}
Type            : AUTOMATION
WindowId        : mw-0c50858d01EXAMPLE
WindowTaskId    : 4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE
```

Exécution d'automatisations à l'aide de différents modèles de sécurité

Cette section inclut des informations sur la façon d'exécuter des automatisations à l'aide de différents modèles de sécurité.

Rubriques

- Exécution d'une automatisation en tant qu'utilisateur actuellement authentifié (p. 364)
- Exécution d'une automatisation à l'aide d'un rôle IAM de service (p. 368)
- Exécution d'une automatisation à l'aide de l'administration déléguée (p. 372)

Exécution d'une automatisation en tant qu'utilisateur actuellement authentifié

Les procédures suivantes décrivent comment exécuter une automatisation qui s'exécute dans le contexte de l'utilisateur AWS Identity and Access Management (IAM) actuel à l'aide de la AWS Systems Manager console , de l' AWS Command Line Interface (AWS CLI) et de Outils AWS pour Windows PowerShell . L'exécution de l'automatisation dans le contexte de l'IAMutilisateur actuel signifie que vous n'avez pas besoin de configurer IAM d'autorisations supplémentaires tant que IAM l'utilisateur a l'autorisation d'exécuter le document Automation et les actions appelées par le document. Si lIAM'utilisateur dispose des autorisations d'administrateur dans IAM , vous avez le droit d'exécuter cette automatisation.

Exécution d'une automatisation en tant qu'utilisateur actuellement authentifié (console)

La procédure suivante décrit comment utiliser la Systems Manager console pour exécuter une automatisation en tant qu'utilisateur actuellement authentifié.

Pour exécuter le document d'Automation en tant qu'utilisateur actuellement authentifié

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Automation (Automatisation), puis Execute automation (Exécuter l'automatisation).
3. Dans la liste Automation document (Document d'automatisation), choisissez un document. Choisissez une ou plusieurs options dans le volet Document categories (Catégories de documents) pour filtrer les documents SSM en fonction de leur objectif. Pour afficher un document dont vous êtes propriétaire, sélectionnez l’onglet Owned by me (M’appartenant). Pour afficher un document partagé avec votre compte, sélectionnez l’onglet Shared with me (Partagé avec moi). Pour afficher tous les documents, sélectionnez l’onglet All documents (Tous les documents).

   Note

   Pour afficher des informations sur un document, sélectionnez son nom.
4. Dans la section Document details (Détails du document), vérifiez que l'option Document version (Version de document) correspond à la version que vous souhaitez exécuter. Le système inclut les options de version suivantes :

- Default version at runtime (Version par défaut lors de l'exécution) : choisissez cette option si le document d'automatisation est mis à jour régulièrement et qu'une nouvelle version par défaut est affectée.
- Latest version at runtime (Dernière version lors de l'exécution) : choisissez cette option si le document Automation est mis à jour régulièrement et que vous souhaitez exécuter la version la plus récemment mise à jour.
- 1 (Default) [1 (Par défaut)] : choisissez cette option pour exécuter la première version du document, qui est la version par défaut.

5. Choisissez Suivant.
6. Dans la section Execution Mode (Mode d'exécution), choisissez Simple execution (Exécution simple).

   Note

   Cette procédure utilise le mode Simple execution (Exécution simple). Cependant, vous pouvez également choisir Rate Control (Contrôle du débit) ou Manual execution (Exécution manuelle) et exécuter le flux de travail Automation en tant qu'utilisateur actuellement authentifié.
7. Dans la section Paramètres d'entrée, spécifiez les entrées obligatoires. Pour exécuter Automation en tant qu'utilisateur actuellement authentifié, ne spécifiez pas de rôle IAM de service pour la valeur AutomationAssumeRole .
8. Sélectionnez Exécuter. La console affiche le statut de l'automatisation.

Exécution d'une automatisation en tant qu'utilisateur actuellement authentifié (ligne de commande)

La procédure suivante décrit comment utiliser l' AWS CLI (sous Linux ou Windows) ou Outils AWS pour PowerShell pour exécuter une automatisation en tant qu'utilisateur actuellement authentifié.

Pour exécuter l'automatisation en tant qu'utilisateur actuellement authentifié

1. Installez et configurez l'AWS CLI ou Outils AWS pour PowerShell si vous ne l'avez pas déjà fait.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).
2. Exécutez la commande suivante pour démarrer une automatisation en tant qu'utilisateur actuellement authentifié.

   Linux

   ```
   aws ssm start-automation-execution \
       --document-name DocumentName \
       --parameters ParametersRequiredByDocument
   ```

   Windows

   ```
   aws ssm start-automation-execution ^
       --document-name DocumentName ^
       --parameters ParametersRequiredByDocument
   ```

   PowerShell

   ```
   Start-SSMAutomationExecution `
       -DocumentName DocumentName `
       -Parameter ParametersRequiredByDocument
   ```

Voici un exemple d'utilisation du document `AWS-RestartEC2Instance` pour redémarrer l'instance EC2 spécifiée.

Linux

```
aws ssm start-automation-execution \
    --document-name "AWS-RestartEC2Instance" \
    --parameters "InstanceId=i-1234567890abcdef0"
```

Windows

```
aws ssm start-automation-execution ^
    --document-name "AWS-RestartEC2Instance" ^
    --parameters "InstanceId=i-1234567890abcdef0"
```

PowerShell

```
Start-SSMAutomationExecution `
    -DocumentName AWS-RestartEC2Instance `
    -Parameter @{"InstanceId"="i-1234567890abcdef0"}
```

Le système renvoie des informations similaires à celles qui suivent.

Linux

```
{
    "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0123456789ab"
}
```

Windows

```
{
    "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0123456789ab"
}
```

PowerShell

```
4105a4fc-f944-11e6-9d32-0123456789ab
```

3. Exécutez la commande suivante pour récupérer le statut du flux de travail Automation.

Linux

```
aws ssm describe-automation-executions \
    --filter "Key=ExecutionId,Values=4105a4fc-f944-11e6-9d32-0123456789ab"
```

Windows

```
aws ssm describe-automation-executions ^
    --filter "Key=ExecutionId,Values=4105a4fc-f944-11e6-9d32-0123456789ab"
```

PowerShell

```
Get-SSMAutomationExecutionList | `
    Where {$_.AutomationExecutionId -eq "4105a4fc-f944-11e6-9d32-0123456789ab"}
```

Le système renvoie des informations similaires à celles qui suivent.

Linux

```
{
    "AutomationExecutionMetadataList": [
        {
            "AutomationExecutionStatus": "InProgress",
            "CurrentStepName": "stopInstances",
            "Outputs": {},
            "DocumentName": "AWS-RestartEC2Instance",
            "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0123456789ab",
            "DocumentVersion": "1",
            "ResolvedTargets": {
                "ParameterValues": [],
                "Truncated": false
            },
            "AutomationType": "Local",
            "Mode": "Auto",
            "ExecutionStartTime": 1564600648.159,
            "CurrentAction": "aws:changeInstanceState",
            "ExecutedBy": "arn:aws:sts::123456789012:assumed-role/Administrator/
Admin",
            "LogFile": "",
            "Targets": []
        }
    ]
}
```

Windows

```
{
    "AutomationExecutionMetadataList": [
        {
            "AutomationExecutionStatus": "InProgress",
            "CurrentStepName": "stopInstances",
            "Outputs": {},
            "DocumentName": "AWS-RestartEC2Instance",
            "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0123456789ab",
            "DocumentVersion": "1",
            "ResolvedTargets": {
                "ParameterValues": [],
                "Truncated": false
            },
            "AutomationType": "Local",
            "Mode": "Auto",
            "ExecutionStartTime": 1564600648.159,
            "CurrentAction": "aws:changeInstanceState",
            "ExecutedBy": "arn:aws:sts::123456789012:assumed-role/Administrator/
Admin",
            "LogFile": "",
            "Targets": []
        }
    ]
}
```

PowerShell

```
AutomationExecutionId       : 4105a4fc-f944-11e6-9d32-0123456789ab
AutomationExecutionStatus   : InProgress
AutomationType              : Local
CurrentAction               : aws:changeInstanceState
CurrentStepName             : startInstances
DocumentName                : AWS-RestartEC2Instance
DocumentVersion             : 1
ExecutedBy                  : arn:aws:sts::123456789012:assumed-role/Administrator/
Admin
ExecutionEndTime            : 1/1/0001 12:00:00 AM
ExecutionStartTime          : 7/31/2019 7:17:28 PM
FailureMessage              :
LogFile                     :
MaxConcurrency              :
MaxErrors                   :
Mode                        : Auto
Outputs                     : {}
ParentAutomationExecutionId :
ResolvedTargets             : Amazon.SimpleSystemsManagement.Model.ResolvedTargets
Target                      :
TargetMaps                  : {}
TargetParameterName         :
Targets                     : {}
```

Exécution d'une automatisation à l'aide d'un rôle IAM de service

Les procédures suivantes décrivent comment utiliser la AWS Systems Manager console , AWS Command Line Interface l' (AWS CLI) et Outils AWS pour Windows PowerShell pour exécuter une automatisation à l'aide d'un rôle de service AWS Identity and Access Management (ou IAM un rôle de responsable) (). Le rôle de service accorde à l'automatisation l'autorisation d'effectuer des actions en votre nom. Il est utile de configurer un rôle de service lorsque vous voulez limiter les autorisations et exécuter des actions avec le principe du moindre privilège. Cela est utile, par exemple, lorsque vous souhaitez restreindre les privilèges d'un utilisateur sur une ressource, telle qu'une instance EC2, mais que vous voulez autoriser l'utilisateur à exécuter une automatisation qui effectue un ensemble spécifique d'actions. Dans ce scénario, vous pouvez créer un rôle de service avec des privilèges élevés et autoriser l'utilisateur à exécuter l'automatisation.

Avant de commencer

Avant d'exécuter les procédures suivantes, vous devez créer le rôle de service IAM et configurer une relation d'approbation pour Automation. Pour plus d'informations, consultez la section Tâche 1 : Créer un rôle de service pour Automation (p. 310).

Exécution d'une automatisation à l'aide d'un rôle IAM de service (console)

La procédure suivante décrit comment utiliser la Systems Manager console pour exécuter une instance d'Automation qui utilise un rôle IAM de service (ou un rôle de responsable).

Pour exécuter une automatisation à l'aide d'un rôle de service

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Automation (Automatisation), puis Execute automation (Exécuter l'automatisation).
3. Dans la liste Automation document (Document d'automatisation), choisissez un document. Choisissez une ou plusieurs options dans le volet Document categories (Catégories de documents) pour filtrer les documents SSM en fonction de leur objectif. Pour afficher un document dont vous êtes propriétaire,

sélectionnez l'onglet Owned by me (M'appartenant). Pour afficher un document partagé avec votre compte, sélectionnez l'onglet Shared with me (Partagé avec moi). Pour afficher tous les documents, sélectionnez l'onglet All documents (Tous les documents).

> **Note**
>
> Pour afficher des informations sur un document, sélectionnez son nom.

4. Dans la section Document details (Détails du document), vérifiez que l'option Document version (Version de document) correspond à la version que vous souhaitez exécuter. Le système inclut les options de version suivantes :
   - Default version at runtime (Version par défaut lors de l'exécution) : choisissez cette option si le document d'automatisation est mis à jour régulièrement et qu'une nouvelle version par défaut est affectée.
   - Latest version at runtime (Dernière version lors de l'exécution) : choisissez cette option si le document Automation est mis à jour régulièrement et que vous souhaitez exécuter la version la plus récemment mise à jour.
   - 1 (Default) [1 (Par défaut)] : choisissez cette option pour exécuter la première version du document, qui est la version par défaut.
5. Choisissez Suivant.
6. Dans la section Execution Mode (Mode d'exécution), choisissez Simple execution (Exécution simple).

   > **Note**
   >
   > Cette procédure utilise le mode Simple execution (Exécution simple). Cependant, vous pouvez également choisir le contrôle du débit, plusieurs comptes et régions, ou une exécution manuelle, et exécuter l'automatisation à l'aide d'un rôle de service.
7. Dans la section Paramètres d'entrée, spécifiez les entrées obligatoires. Dans la zone Automation Assume Role (Rôle responsable d'automatisation), collez l'ARN du rôle de service IAM.
8. Sélectionnez Exécuter. La console affiche le statut de l'automatisation.

Exécution d'une automatisation à l'aide d'un rôle de IAM service (ligne de commande)

La procédure suivante décrit comment utiliser l' AWS CLI (sous Linux ou Windows) ou Outils AWS pour PowerShell exécuter une instance d'Automation qui utilise un rôle IAM de service (ou un rôle responsable).

Pour exécuter une automatisation à l'aide d'un rôle de service

1. Installez et configurez l'AWS CLI ou Outils AWS pour PowerShell si vous ne l'avez pas déjà fait.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).
2. Exécutez la commande suivante pour démarrer une automatisation qui utilise un rôle IAM de service .

   Linux

   ```
   aws ssm start-automation-execution \
       --document-name DocumentName \
       --parameters
    "ParametersRequiredByDocument","AutomationAssumeRole=arn:aws:iam::123456789012:role/
   AmazonSSMAutomationRole"
   ```

   Windows

   ```
   aws ssm start-automation-execution ^
       --document-name DocumentName ^
   ```

```
    --parameters
 "ParametersRequiredByDocument","AutomationAssumeRole=arn:aws:iam::123456789012:role/
AmazonSSMAutomationRole"
```

PowerShell

```
Start-SSMAutomationExecution `
    -DocumentName DocumentName `
    -Parameter @{
    "ParametersRequiredByDocument"="ParameterValues";
    "AutomationAssumeRole"="arn:aws:iam::123456789012:role/
AmazonSSMAutomationRole"}
```

Voici un exemple d'utilisation du document `AWS-RestartEC2Instance` pour redémarrer l'instance EC2 spécifiée à l'aide du rôle de service IAM `AmazonSSMAutomationRole`.

Linux

```
aws ssm start-automation-execution \
    --document-name "AWS-RestartEC2Instance" \
    --parameters
 "InstanceId=i-1234567890abcdef0","AutomationAssumeRole=arn:aws:iam::123456789012:role/
AmazonSSMAutomationRole"
```

Windows

```
aws ssm start-automation-execution ^
    --document-name "AWS-RestartEC2Instance" ^
    --parameters
 "InstanceId=i-1234567890abcdef0","AutomationAssumeRole=arn:aws:iam::123456789012:role/
AmazonSSMAutomationRole"
```

PowerShell

```
Start-SSMAutomationExecution `
    -DocumentName "AWS-RestartEC2Instance" `
    -Parameter @{
    "InstanceId"="i-1234567890abcdef0";
    "AutomationAssumeRole"="arn:aws:iam::123456789012:role/AWS-SystemsManager-
AutomationAdministrationRole"}
```

Le système renvoie des informations similaires à celles qui suivent.

Linux

```
{
    "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0123456789ab"
}
```

Windows

```
{
    "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0123456789ab"
}
```

PowerShell

```
4105a4fc-f944-11e6-9d32-0123456789ab
```

3. Exécutez la commande suivante pour récupérer le statut de l'automatisation.

Linux

```
aws ssm describe-automation-executions \
    --filter "Key=ExecutionId,Values=4105a4fc-f944-11e6-9d32-0123456789ab"
```

Windows

```
aws ssm describe-automation-executions ^
    --filter "Key=ExecutionId,Values=4105a4fc-f944-11e6-9d32-0123456789ab"
```

PowerShell

```
Get-SSMAutomationExecutionList | `
    Where {$_.AutomationExecutionId -eq "4105a4fc-f944-11e6-9d32-0123456789ab"}
```

Le système renvoie des informations similaires à celles qui suivent.

Linux

```
{
    "AutomationExecutionMetadataList": [
        {
            "AutomationExecutionStatus": "InProgress",
            "CurrentStepName": "stopInstances",
            "Outputs": {},
            "DocumentName": "AWS-RestartEC2Instance",
            "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0123456789ab",
            "DocumentVersion": "1",
            "ResolvedTargets": {
                "ParameterValues": [],
                "Truncated": false
            },
            "AutomationType": "Local",
            "Mode": "Auto",
            "ExecutionStartTime": 1564600648.159,
            "CurrentAction": "aws:changeInstanceState",
            "ExecutedBy": "arn:aws:sts::123456789012:assumed-role/Administrator/
Admin",
            "LogFile": "",
            "Targets": []
        }
    ]
}
```

Windows

```
{
    "AutomationExecutionMetadataList": [
        {
            "AutomationExecutionStatus": "InProgress",
            "CurrentStepName": "stopInstances",
```

```
            "Outputs": {},
            "DocumentName": "AWS-RestartEC2Instance",
            "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0123456789ab",
            "DocumentVersion": "1",
            "ResolvedTargets": {
                "ParameterValues": [],
                "Truncated": false
            },
            "AutomationType": "Local",
            "Mode": "Auto",
            "ExecutionStartTime": 1564600648.159,
            "CurrentAction": "aws:changeInstanceState",
            "ExecutedBy": "arn:aws:sts::123456789012:assumed-role/Administrator/
Admin",
            "LogFile": "",
            "Targets": []
        }
    ]
}
```

PowerShell

```
AutomationExecutionId       : 4105a4fc-f944-11e6-9d32-0123456789ab
AutomationExecutionStatus   : InProgress
AutomationType              : Local
CurrentAction               : aws:changeInstanceState
CurrentStepName             : startInstances
DocumentName                : AWS-RestartEC2Instance
DocumentVersion             : 1
ExecutedBy                  : arn:aws:sts::123456789012:assumed-role/Administrator/
Admin
ExecutionEndTime            : 1/1/0001 12:00:00 AM
ExecutionStartTime          : 7/31/2019 7:17:28 PM
FailureMessage              :
LogFile                     :
MaxConcurrency              :
MaxErrors                   :
Mode                        : Auto
Outputs                     : {}
ParentAutomationExecutionId :
ResolvedTargets             : Amazon.SimpleSystemsManagement.Model.ResolvedTargets
Target                      :
TargetMaps                  : {}
TargetParameterName         :
Targets                     : {}
```

Pour plus d'exemples sur l'utilisation de Systems Manager Automation, veuillez consulter Procédures d'automatisation (p. 632). Pour de plus amples informations sur comment débuter avec Automation, veuillez consulter Configuration d'Automation (p. 308).

Exécution d'une automatisation à l'aide de l'administration déléguée

Lorsque vous exécutez une automatisation, par défaut, l'automatisation s'exécute dans le contexte de l'utilisateur AWS Identity and Access Management (IAM) qui a lancé l'exécution. Cela signifie, par exemple, que si votre compte IAM utilisateur dispose d'autorisations d'administrateur, l'automatisation s'exécute avec des autorisations d'administrateur et un accès total aux ressources configurées par l'automatisation. En tant que bonne pratique de sécurité, nous vous recommandons d'exécuter l'automatisation à l'aide d'un rôle de IAM service (également appelé rôle assumé) qui est configuré avec la stratégie AmazonSSMAutomationRole gérée. L'utilisation d'un rôle IAM de service pour exécuter l'automatisation est appelée administration déléguée.

Lorsque vous utilisez un rôle de service, l'automatisation est autorisée à s'exécuter sur les AWS ressources, mais l'utilisateur qui a exécuté l'automatisation dispose d'un accès restreint (ou d'aucun accès) à ces ressources. Par exemple, vous pouvez configurer un rôle de service et l'utiliser avec Automation afin de redémarrer une ou plusieurs instances EC2. L'automatisation redémarre les instances, mais le rôle de service n'autorise pas l'utilisateur à accéder à ces instances.

Vous pouvez spécifier un rôle de service lors de l'exécution lorsque vous exécutez une automatisation, ou vous pouvez créer des documents Automation personnalisés et spécifier le rôle de service directement dans le document. Si vous spécifiez un rôle de service au moment de l'exécution ou dans un document d'Automation, le service est alors exécuté dans le contexte du rôle de service spécifié. Si vous ne spécifiez pas de rôle de service, le système crée une session temporaire dans le contexte de l'utilisateur et exécute l'automatisation.

> **Note**
>
> Vous devez spécifier un rôle de service pour l'automatisation que vous prévoyez d'exécuter pendant plus de 12 heures. Si vous démarrez une automatisation de longue durée dans le contexte d'un utilisateur, la session temporaire de l'utilisateur expire au bout de 12 heures.

L'administration déléguée garantit un niveau de sécurité élevé et le contrôle de vos ressources AWS. Elle permet également une expérience d'audit améliorée parce que des actions sont effectuées par rapport à vos ressources par un rôle de service central au lieu de plusieurs comptes IAM.

Afin d'illustrer la façon dont l'administration déléguée peut fonctionner dans une organisation, cette rubrique détaille les tâches suivantes comme si elles étaient exécutées par trois personnes différentes :

- Créez un test de compte IAM utilisateur appelé AutomationRestrictedOperator (Administrateur).
- Créer un rôle de service IAM pour Automation (administrateur).
- Créer un document Automation simple (basé sur un document Automation préexistant) qui spécifie le rôle de service (auteur du document SSM).
- Exécutez l'automatisation en tant qu'utilisateur test (opérateur restreint).

Dans certaines organisations, ces trois tâches sont réalisées par la même personne, mais ici, l'identification des différents rôles peut vous aider à comprendre la façon dont l'administration déléguée améliore la sécurité dans les organisations complexes.

> **Important**
>
> La bonne pratique en matière de sécurité consiste à toujours utiliser un rôle de service pour exécuter des automatisations, même si vous êtes un administrateur qui effectue toutes ces tâches.

Les procédures de cette section proposent des liens vers des rubriques dans d'autres guides AWS ou d'autres rubriques Systems Manager. Nous vous recommandons d'ouvrir les liens vers d'autres rubriques dans un nouvel onglet de votre navigateur web afin de conserver l'endroit où vous vous trouviez dans cette rubrique.

Rubriques

- Créer un compte utilisateur test (p. 373)
- Créer un rôle de service IAM pour Automation (p. 374)
- Créer un document d'automatisation personnalisé (p. 374)
- Exécuter le document Automation personnalisé (p. 375)

Créer un compte utilisateur test

Cette section décrit la création d'un compte utilisateur test IAM avec des autorisations restreintes. Le jeu d'autorisations permet à l'utilisateur d'exécuter des automatisations, mais l'utilisateur n'a pas accès

aux AWS ressources ciblées par Automation. L'opérateur peut également afficher les résultats des automatisations. Vous commencez par créer la stratégie d'autorisations IAM personnalisées, puis vous créez le compte utilisateur et vous lui attribuez les autorisations.

Créer un utilisateur test IAM

1. Créez une stratégie d'autorisation nommée OperatorRestrictedPermissions . Pour plus d'informations sur la création d'une nouvelle stratégie d'IAMautorisations, consultez Créer une stratégie IAM (console) dans le IAM Guide de l'utilisateur . Créez la stratégie sur l'onglet JSON et spécifiez les ensembles d'autorisations suivants.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "ssm:DescribeAutomationExecutions",
            "ssm:DescribeAutomationStepExecutions",
            "ssm:DescribeDocument",
            "ssm:GetAutomationExecution",
            "ssm:GetDocument",
            "ssm:ListDocuments",
            "ssm:ListDocumentVersions",
            "ssm:StartAutomationExecution"
         ],
         "Resource":"*"
      }
   ]
}
```

2. Créez un nouveau compte IAM utilisateur nommé AutomationRestrictedOperator . Pour plus d'informations sur la création dIAM'un nouvel utilisateur , consultez Création d'utilisateurs IAM (console) dans le IAM Guide de l'utilisateur . Lorsque vous y êtes invité, choisissez Attach existing policies directly (Attacher directement les stratégies existantes), puis choisissez la stratégie que vous venez de créer.
3. Notez le nom d'utilisateur, le mot de passe et le lien vers le Console login link (lien de connexion à la console). Vous vous connecterez à ce compte plus tard dans cette rubrique.

Créer un rôle de service IAM pour Automation

La procédure suivante propose des liens vers d'autres rubriques afin de vous aider à créer le rôle de service et à configurer Automation de façon à ce qu'elle approuve ce rôle.

Pour créer le rôle de service et activer Automation pour l'approuver

1. Créer le rôle de service Automation. Pour plus d'informations, consultez la section Tâche 1 : Créer un rôle de service pour Automation (p. 310).
2. Notez l'ARN (Amazon Resource Name) du rôle de service. Vous indiquerez cet ARN au cours de la procédure suivante.

Créer un document d'automatisation personnalisé

Cette section décrit comment créer un document Automation personnalisé qui redémarre les instances EC2. AWS fournit un document SSM par défaut pour les instances de redémarrage appelées AWS-RestartEC2Instance. La procédure suivante copie le contenu de ce document dans le but de vous montrer comment entrer le rôle de service dans un document lorsque vous créez vos propres rôles. En spécifiant le service directement dans le document, l'utilisateur exécutant le document n'a pas besoin d'autorisations

iam:PassRole. Sans autorisations iam:PassRole, l'utilisateur ne peut pas utiliser le rôle de service ailleurs dans AWS.

Pour créer un document d'Automation personnalisé

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Documents dans le panneau de navigation.
3. Choisissez Create document (Créer un document).
4. Dans le champ Nom, saisissez un nom pour le document, par exemple Restart-EC2InstanceDemo.
5. Dans la liste Types de document, choisissez document d'Automation.
6. Dans la section Contenu, choisissez JSON, puis collez le contenu suivant. – Remplacez AssumeRoleARN par l'ARN du rôle de service que vous avez créé dans la procédure précédente.

```
{
  "description": "Restart EC2 instances(s)",
  "schemaVersion": "0.3",
  "assumeRole": "AssumeRoleARN",
  "parameters": {
    "InstanceId": {
      "type": "StringList",
      "description": "(Required) EC2 Instance to restart"
    }
  },
  "mainSteps": [
    {
      "name": "stopInstances",
      "action": "aws:changeInstanceState",
      "inputs": {
        "InstanceIds": "{{ InstanceId }}",
        "DesiredState": "stopped"
      }
    },
    {
      "name": "startInstances",
      "action": "aws:changeInstanceState",
      "inputs": {
        "InstanceIds": "{{ InstanceId }}",
        "DesiredState": "running"
      }
    }
  ]
}
```

7. Choisissez Create document (Créer un document).

Exécuter le document Automation personnalisé

La procédure suivante décrit l'exécution du document que vous venez de créer à l'aide du rôle d'opérateur restreint créé précédemment dans cette rubrique. L'utilisateur peut exécuter le document créé précédemment parce que ses autorisations de compte IAM lui permettent de consulter et d'exécuter le document. L'utilisateur ne peut toutefois pas se connecter aux instances que vous redémarrerez avec cette automatisation.

1. Dans l' https://console.aws.amazon.com/ec2/ , copiez l'instance IDs d'une ou de plusieurs instances que vous souhaitez redémarrer à l'aide de l'automatisation suivante.
2. Déconnectez-vous de l'AWS Management Console, puis reconnectez-vous en utilisant le Console login link (Lien de connexion à la console) au compte d'utilisateur de test que vous avez copié précédemment.
3. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
4. Dans le panneau de navigation, choisissez Automatisation.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Automation.
5. Choisissez Execute automation (Exécuter l'automatisation).
6. Choisissez le document d'Automation personnalisé que vous avez créé précédemment au cours de cette rubrique.
7. Dans la section Document details (Détails du document), vérifiez que l'option Document version (Version du document) est définie sur 1 (Default) (1 (Par défaut)).
8. Choisissez Nex t(Suivant).
9. Dans la section Mode Execution mode (Mode d'exécution), choisissez Simple execution (Exécution simple).
10. Dans la section Paramètres d'entrée, tapez une ou plusieurs instances IDs que vous souhaitez redémarrer, puis choisissez Exécuter.

Les détails d'exécution décrivent l'état de l'automatisation. L'étape 1 interrompt les instances. L'étape 2 démarre les instances.

Exécution d'automatisations dans plusieurs AWS comptes et régions

Vous pouvez exécuter des AWS Systems Manager automatisations dans plusieurs AWS régions et AWS comptes ou unités AWS d'organisation à partir d'un compte de gestion d'automatisation. Des automatisations exécutées dans plusieurs régions et comptes ou OUs réduit le temps nécessaire pour administrer vos AWS ressources , tout en renforçant la sécurité de votre environnement informatique.

Par exemple, vous pouvez implémenter de manière centralisée les correctifs et les mises à jour de sécurité, corriger les écarts de conformité dans les configurations de VPC ou les stratégies de compartiment S3, et gérer les ressources, telles que des instances EC2, à l'échelle. Le graphique suivant illustre un exemple d'utilisateur qui exécute le document AWS-RestartEC2Instances dans plusieurs régions et comptes à partir d'un compte de gestion Automation. L'automatisation localise les instances en utilisant les balises spécifiées dans les régions et comptes spécifiés.

> **Note**
>
> Lorsque vous exécutez une automatisation dans plusieurs régions et comptes, vous ciblez les ressources à l'aide de balises ou du nom d'un groupe de AWS ressources . Le groupe de ressources doit exister dans chaque compte cible et région, et le nom du groupe de ressources doit être le même dans chaque compte cible et région. L'automatisation ne peut pas s'exécuter sur les ressources qui n'ont pas la balise spécifiée ou qui ne sont pas incluses dans le groupe de ressources spécifié.

Multi-Region and Account Automation to Restart EC2 Instances Tagged with: Key=Env,Value=Prod

> **Important**
>
> Votre compte est facturé pour l'exécution d'automatisations dans plusieurs régions et comptes. Les étapes d'exécution dans plusieurs régions et plusieurs comptes sont considérées comme des étapes spéciales. Aucune limite spécifique ne s'applique aux étapes spéciales, mais votre compte est facturé pour chaque étape traitée par Systems Manager. Pour plus d'informations, consultez la page de Tarification AWS Systems Manager.

Fonctionnement

L'exécution d'automatisations dans plusieurs régions et comptes OUs fonctionne comme suit :

1. Vérifiez que toutes les ressources sur lesquelles vous souhaitez exécuter l'automatisation, dans toutes les régions et comptes ou OUs , utilisent des balises identiques. Si ce n'est pas le cas, vous pouvez les ajouter à un groupe de ressources AWS et cibler ce groupe. Pour de plus amples informations, veuillez consulter la documentation relative à la présentation d'Groupes de ressources AWS.
2. Connectez-vous au compte AWS Identity and Access Management (IAM) que vous souhaitez configurer comme compte principal Automation.
3. Utilisez la procédure décrite dans cette rubrique pour créer le rôle IAM d'automatisation nommé AWS-SystemsManager-AutomationExecutionRole. Ce rôle donne à l'utilisateur l'autorisation d'exécuter des flux de travail d'automatisation.
4. Utilisez la procédure décrite dans cette rubrique pour créer le second rôle IAM nommé AWS-SystemsManager-AutomationAdministrationRole. Ce rôle donne à l'utilisateur l'autorisation d'exécuter des flux de travail d'automatisation dans plusieurs AWS comptes et OUs .
5. Choisissez le document Automation, les régions et les comptes OUs ou l'endroit où vous souhaitez exécuter l'automatisation.

   > **Note**
   >
   > Les automatisations ne s'exécutent pas de manière récursive via OUs . Veillez à ce que l'unité d'organisation cible contienne les comptes souhaités.

6. Exécutez l'automatisation. Lors de l'exécution d'automatisations dans plusieurs régions, comptes ou OUs , l'automatisation que vous exécutez à partir du compte principal démarre les automatisations enfants dans chacun des comptes de destination. L'automatisation dans le compte principal comporte `aws:executeAutomation` des étapes pour chacun des comptes de destination.
7. Utilisez les actions GetAutomationExecution d'API DescribeAutomationStepExecutions , et à partir de la console ou de l' DescribeAutomationExecutions pour surveiller la progression du flux de travail.AWS Systems ManagerAWS CLI La sortie des étapes pour l'automatisation de votre compte principal sera l`AutomationExecutionId`'automatisation enfant. Pour afficher la sortie des automatisations enfants créées dans vos comptes de destination, veillez à spécifier le compte, la région et `AutomationExecutionId` dans votre demande appropriés.

Configuration des autorisations de compte de gestion pour l'automatisation dans plusieurs régions et plusieurs comptes

Utilisez la procédure suivante pour créer les IAM rôles requis pour Automation dans plusieurs régions et plusieurs comptes avec Systems Manager .AWS CloudFormation Cette procédure décrit la création du rôle AWS-SystemsManager-AutomationExecutionRole. Vous devez créer ce rôle dans chaque compte que vous souhaitez cibler pour exécuter des automatisations dans plusieurs régions et plusieurs comptes.

Cette procédure décrit également la création du rôle AWS-SystemsManager-AutomationAdministrationRole. Vous devez uniquement créer ce rôle dans le compte de gestion d'Automation.

Pour créer les IAM rôles requis pour les automatisations dans plusieurs régions et plusieurs comptes à l'aide d'AWS CloudFormation

1. Téléchargez et décompressez le dossier AWS-SystemsManager-AutomationExecutionRole.zip. Ce dossier inclut le fichier modèle AWS-SystemsManager-AutomationExecutionRole.json AWS CloudFormation.

   > **Note**
   >
   > Nous vous recommandons de ne pas remplacer le nom du rôle tel que spécifié dans le modèle par autre chose que `AWS-SystemsManager-AutomationExecutionRole` . Sinon, vos automatisations dans plusieurs régions et plusieurs comptes risquent d'échouer.
2. Ouvrez la console AWS CloudFormation à l'adresse https://console.aws.amazon.com/cloudformation.
3. Choisissez Créer une pile.
4. Dans la section Choisir un modèle, choisissez Télécharger un modèle sur Amazon S3.
5. Choisissez Browse (Parcourir), puis choisissez le fichier modèle AWS CloudFormation `AWS-SystemsManager-AutomationExecutionRole.json`.
6. Choisissez Nex t(Suivant).
7. Dans la section Spécifier les détails, entrez un nom de pile dans le champ Nom de la pile.
8. Dans la section Parameters (Paramètres), dans le MasterAccountID champ, entrez l'ID du compte que vous souhaitez utiliser pour exécuter des automatisations dans plusieurs régions et plusieurs comptes.
9. Choisissez Nex t(Suivant).
10. Sur la page Options, entrez des valeurs pour les options que vous souhaitez utiliser. Choisissez Nex t(Suivant).
11. Faites défiler la page Vérification vers le bas et choisissez l'option Je comprends qu'AWS CloudFormation peut créer des ressources IAM.
12. Sélectionnez Create.

    AWS CloudFormation affiche le statut CREATE_IN_PROGRESS pendant environ trois minutes. L'état passe à CREATE_COMPLETE.

13. Répétez cette procédure dans chaque compte que vous souhaitez cibler pour exécuter des automatisations dans plusieurs régions et plusieurs comptes.
14. Téléchargez le dossier AWS-SystemManager-AutomationAdministrationRole.zip et répétez cette procédure pour le rôle AWS-SystemManager-AutomationAdministrationRole. Vous devez uniquement créer le rôle AWS-SystemManager-AutomationAdministrationRole dans le compte de gestion d'Automation.

    Note

    LIAM'utilisateur ou le rôle que vous utilisez pour exécuter une automatisation sur plusieurs régions ou plusieurs comptes doit disposer de l`iam:PassRole`'autorisation pour le `AWS-SystemManager-AutomationAdministrationRole` rôle. Nous vous recommandons de ne pas remplacer le nom du rôle tel que spécifié dans le modèle par autre chose que `AWS-SystemsManager-AutomationAdministrationRole` . Sinon, vos automatisations dans plusieurs régions et plusieurs comptes risquent d'échouer.

Exécuter une automatisation dans plusieurs comptes et régions (console)

La procédure suivante décrit comment utiliser la Systems Manager console pour exécuter une automatisation dans plusieurs régions et comptes à partir du compte de gestion Automation.

Avant de commencer

Avant d'exécuter la procédure suivante, notez les informations suivantes :

- AWSou l'IDsendroit où vous souhaitez exécuter l'automatisation.OUs
- Régions prises en charge par Systems Manager où vous souhaitez exécuter l'automatisation.
- La clé de balise et la valeur de balise, ou le nom du groupe de ressources, où vous souhaitez exécuter l'automatisation.

Pour exécuter une automatisation dans plusieurs comptes et régions

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Automation (Automatisation), puis Execute automation (Exécuter l'automatisation).
3. Dans la liste Automation document (Document d'automatisation), choisissez un document. Choisissez une ou plusieurs options dans le volet Document categories (Catégories de documents) pour filtrer les documents SSM en fonction de leur objectif. Pour afficher un document dont vous êtes propriétaire, sélectionnez l’onglet Owned by me (M’appartenant). Pour afficher un document partagé avec votre compte, sélectionnez l’onglet Shared with me (Partagé avec moi). Pour afficher tous les documents, sélectionnez l’onglet All documents (Tous les documents).

    Note

    Pour afficher des informations sur un document, sélectionnez son nom.
4. Dans la section Document details (Détails du document), vérifiez que l'option Document version (Version de document) correspond à la version que vous souhaitez exécuter. Le système inclut les options de version suivantes :

    - Default version at runtime (Version par défaut lors de l'exécution) : choisissez cette option si le document d'automatisation est mis à jour régulièrement et qu'une nouvelle version par défaut est affectée.
    - Latest version at runtime (Dernière version lors de l'exécution) : choisissez cette option si le document Automation est mis à jour régulièrement et que vous souhaitez exécuter la version la plus récemment mise à jour.

- 1 (Default) [1 (Par défaut)] : choisissez cette option pour exécuter la première version du document, qui est la version par défaut.

5. Choisissez Suivant.
6. Sur la page Execute automation document (Exécuter le document d'Automation) , choisissez Multi-account and Region (Compte et région multiples).
7. Dans la section Comptes et régions de destination, utilisez le champ Comptes et unités d'organisation (UO) pour spécifier les différents AWS comptes ou unités d'organisation (UO) AWS dans lesquels vous souhaitez exécuter l'automatisation. Séparez les comptes par une virgule.OUs
8. Utilisez la liste des régions AWS pour choisir une ou plusieurs régions dans lesquelles vous souhaitez exécuter l'automatisation.
9. Utilisez les options Multi-Region et Account Rate Control pour restreindre l'automatisation à un nombre limité de comptes s'exécutant dans un nombre limité de régions. Ces options ne limitent pas le nombre de AWS ressources pouvant exécuter les automatisations.

   a. Dans la section Simultanéité de l'emplacement (paire compte-région), choisissez une option pour limiter le nombre d'automatisations qui peuvent s'exécuter dans plusieurs comptes et régions en même temps. Par exemple, si vous choisissez d'exécuter une automatisation dans cinq (5) AWS comptes, qui sont situés dans quatre (4) AWS régions, Systems Manager exécute des automatisations dans un total de 20 paires compte-région. Vous pouvez utiliser cette option pour spécifier un nombre absolu, tel que **2** , afin que l'automatisation s'exécute uniquement dans deux paires compte-région en même temps. Vous pouvez également spécifier un pourcentage de paires compte-région qui peuvent s'exécuter en même temps. Par exemple, avec 20 paires compte-région, si vous spécifiez 20 %, l'automatisation s'exécute simultanément dans un maximum de cinq (5) paires compte-région.

      - Choisissez des cibles pour entrer un nombre absolu de paires compte-région pouvant exécuter l'automatisation simultanément.
      - Choisissez le pourcentage pour entrer un pourcentage du nombre total de paires compte-région pouvant exécuter l'automatisation simultanément.

   b. Dans la section Error threshold (Seuil d'erreurs), choisissez une option :

      - Choisissez erreurs pour indiquer un nombre absolu d'erreurs autorisées avant qu'Automation ne cesse d'envoyer le flux de travail à d'autres ressources.
      - Choisissez pourcentage pour indiquer un pourcentage d'erreurs autorisées avant qu'Automation ne cesse d'envoyer le flux de travail à d'autres ressources.

10. Dans la section Targets (Cibles), choisissez la façon dont vous souhaitez cibler les ressources AWS où vous souhaitez exécuter l'automatisation. Ces options sont obligatoires.

    a. Utilisez la liste Parameter (Paramètre) pour choisir un paramètre. Les éléments de la liste Parameter (Paramètre) sont déterminés par les paramètres du document d'automatisation que vous avez sélectionnés au début de cette procédure. En choisissant un paramètre, vous définissez le type de ressource sur lequel le flux de travail d'automatisation s'exécutera.

    b. Utilisez la liste Targets (Cibles) pour choisir la façon dont vous souhaitez cibler les ressources.

       i. Si vous avez choisi de cibler des ressources à l'aide de valeurs de paramètre, entrez la valeur de paramètre que vous avez sélectionnée dans la section Input parameters (Paramètres d'entrée).

       ii. Si vous choisissez de cibler les ressources à l'aide de Groupes de ressources AWS, choisissez le nom du groupe dans la liste Resource Group (Groupe de ressources).

       iii. Si vous choisissez de cibler des ressources à l'aide de balises, entrez la clé de balise et (éventuellement) la valeur de balise dans les champs fournis. Choisissez Ajouter.

       iv. Si vous voulez exécuter un playbook d'automatsation sur toutes les instances du compte AWS actuel et de la région, alors sélectionnez All instances (Toutes les instances).

11. Dans la section Input parameters (Paramètres d'entrée), spécifiez les entrées obligatoires. Le cas échéant, vous pouvez choisir un rôle de service IAM dans la liste AutomationAssumeRole.

    Note

    Vous n'avez pas toujours besoin de choisir certaines options dans la section Paramètres d'entrée. Cela est dû au fait que vous avez ciblé des ressources dans plusieurs régions et comptes à l'aide de balises ou d'un groupe de ressources. Par exemple, si vous avez choisi le document AWS-RestartEC2Instance, vous n'avez pas besoin de spécifier ou de choisir l'instance IDs dans la section Paramètres d'entrée. L'automatisation localise les instances à redémarrer en utilisant les balises que vous avez spécifiées.

12. Utilisez les options de la section Rate control (Contrôle du débit) pour restreindre le nombre de ressources AWS qui peuvent exécuter l'automatisation au sein de chaque paire compte-région.

    Dans la section Concurrency (Simultanéité), choisissez une option :

    - Choisissez targets (cibles) pour entrer un nombre absolu de cibles pouvant exécuter le flux de travail Automation simultanément.
    - Choisissez percentage (pourcentage) pour indiquer un pourcentage de l'ensemble de cibles pouvant exécuter le flux de travail Automation simultanément.

13. Dans la section Error threshold (Seuil d'erreurs), choisissez une option :

    - Choisissez errors (erreurs) pour indiquer un nombre absolu d'erreurs autorisées avant que l'automatisation ne cesse d'envoyer le flux de travail à d'autres ressources.
    - Choisissez percentage (pourcentage) pour indiquer un pourcentage d'erreurs autorisées avant que l'automatisation ne cesse d'envoyer le flux de travail à d'autres ressources.

14. Sélectionnez Exécuter.

Exécuter Automation dans plusieurs comptes et régions AWS (ligne de commande)

La procédure suivante décrit comment utiliser l' AWS CLI (sous Linux ou Windows) ou Outils AWS pour PowerShell pour exécuter une automatisation dans plusieurs régions et comptes à partir du compte de gestion Automation.

Avant de commencer

Avant d'exécuter la procédure suivante, notez les informations suivantes :

- AWSou l'IDsendroit où vous souhaitez exécuter l'automatisation.OUs
- Régions prises en charge par Systems Manager où vous souhaitez exécuter l'automatisation.
- La clé de balise et la valeur de balise, ou le nom du groupe de ressources, où vous souhaitez exécuter l'automatisation.

Pour exécuter une automatisation dans plusieurs comptes et régions

1. Installez et configurez l'AWS CLI ou Outils AWS pour PowerShell si vous ne l'avez pas déjà fait.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Utilisez le format suivant pour créer une commande afin d'exécuter une automatisation dans plusieurs régions et comptes.

Linux

```
aws ssm start-automation-execution \
    --document-name name_of_Automation_document \
    --parameters
 AutomationAssumeRole=arn:aws:iam::Automation_management_account_ID:role/AWS-
SystemsManager-AutomationAdministrationRole \
    --target-parameter-name parameter_name (required) \
    --targets Key=tag_key,Values=tag_value \
    --target-locations
 Accounts=account_ID_1,account_ID_2,account_ID_3,Regions=Region_1,Region_2,ExecutionRoleName=AW
SystemsManager-AutomationExecutionRole
```

Windows

```
aws ssm start-automation-execution ^
    --document-name name_of_Automation_document ^
    --parameters
 AutomationAssumeRole=arn:aws:iam::Automation_management_account_ID:role/AWS-
SystemsManager-AutomationAdministrationRole ^
    --target-parameter-name parameter_name (required) ^
    --targets Key=tag_key,Values=tag_value ^
    --target-locations
 Accounts=account_ID_1,account_ID_2,account_ID_3,Regions=Region_1,Region_2,ExecutionRoleName=AW
SystemsManager-AutomationExecutionRole
```

PowerShell

```
$Targets = New-Object Amazon.SimpleSystemsManagement.Model.Target
$Targets.Key = "target_key"
$Targets.Values = "target_value"

Start-SSMAutomationExecution `
    -DocumentName "name_of_Automation_document" `
    -Parameter @{
    "AutomationAssumeRole"="arn:aws:iam::Automation_management_account_ID:role/AWS-
SystemsManager-AutomationAdministrationRole" } `
    -TargetParameterName "parameter_name (required)" `
    -Target $Targets `
    -TargetLocation @{
    "Accounts"="account_ID_1","account_ID_2","account_ID_3";
    "Regions"="Region_1","Region_2";
    "ExecutionRoleName"="AWS-SystemsManager-AutomationExecutionRole" }
```

Voici quelques exemples.

Exemple 1 : Cet exemple redémarre les instances EC2 des `123456789012` comptes `987654321098` et , qui sont situés dans les `us-east-2` régions `us-west-1` et . Les instances doivent être balisées avec la valeur de paire de clés de balise `Env-PROD`.

Linux

```
aws ssm start-automation-execution \
    --document-name AWS-RestartEC2Instance \
    --parameters AutomationAssumeRole=arn:aws:iam::123456789012:role/AWS-
SystemsManager-AutomationAdministrationRole \
    --target-parameter-name InstanceId \
    --targets Key=tag:Env,Values=PROD \
```

```
    --target-locations Accounts=123456789012,987654321098,Regions=us-east-2,us-
west-1,ExecutionRoleName=AWS-SystemsManager-AutomationExecutionRole
```

Windows

```
aws ssm start-automation-execution ^
    --document-name AWS-RestartEC2Instance ^
    --parameters AutomationAssumeRole=arn:aws:iam::123456789012:role/AWS-
SystemsManager-AutomationAdministrationRole ^
    --target-parameter-name InstanceId ^
    --targets Key=tag:Env,Values=PROD ^
    --target-locations Accounts=123456789012,987654321098,Regions=us-east-2,us-
west-1,ExecutionRoleName=AWS-SystemsManager-AutomationExecutionRole
```

PowerShell

```
$Targets = New-Object Amazon.SimpleSystemsManagement.Model.Target
$Targets.Key = "tag:Env"
$Targets.Values = "PROD"

Start-SSMAutomationExecution `
    -DocumentName "AWS-RestartEC2Instance" `
    -Parameter @{
    "AutomationAssumeRole"="arn:aws:iam::123456789012:role/AWS-SystemsManager-
AutomationAdministrationRole" } `
    -TargetParameterName "InstanceId" `
    -Target $Targets `
    -TargetLocation @{
    "Accounts"="123456789012","987654321098";
    "Regions"="us-east-2","us-west-1";
    "ExecutionRoleName"="AWS-SystemsManager-AutomationExecutionRole" }
```

Exemple 2 : Cet exemple redémarre les instances EC2 des `123456789012` comptes `987654321098` et , qui sont situés dans la `eu-central-1` région . Les instances doivent être membres du groupe de ressources AWS `prod-instances`.

Linux

```
aws ssm start-automation-execution \
    --document-name AWS-RestartEC2Instance \
    --parameters AutomationAssumeRole=arn:aws:iam::123456789012:role/AWS-
SystemsManager-AutomationAdministrationRole \
    --target-parameter-name InstanceId \
    --targets Key=ResourceGroup,Values=prod-instances \
    --target-locations Accounts=123456789012,987654321098,Regions=eu-
central-1,ExecutionRoleName=AWS-SystemsManager-AutomationExecutionRole
```

Windows

```
aws ssm start-automation-execution ^
    --document-name AWS-RestartEC2Instance ^
    --parameters AutomationAssumeRole=arn:aws:iam::123456789012:role/AWS-
SystemsManager-AutomationAdministrationRole ^
    --target-parameter-name InstanceId ^
    --targets Key=ResourceGroup,Values=prod-instances ^
    --target-locations Accounts=123456789012,987654321098,Regions=eu-
central-1,ExecutionRoleName=AWS-SystemsManager-AutomationExecutionRole
```

PowerShell

```
$Targets = New-Object Amazon.SimpleSystemsManagement.Model.Target
$Targets.Key = "ResourceGroup"
$Targets.Values = "prod-instances"

Start-SSMAutomationExecution `
    -DocumentName "AWS-RestartEC2Instance" `
    -Parameter @{
    "AutomationAssumeRole"="arn:aws:iam::123456789012:role/AWS-SystemsManager-
AutomationAdministrationRole" } `
    -TargetParameterName "InstanceId" `
    -Target $Targets `
    -TargetLocation @{
    "Accounts"="123456789012","987654321098";
    "Regions"="eu-central-1";
    "ExecutionRoleName"="AWS-SystemsManager-AutomationExecutionRole" }
```

Exemple 3 : Cet exemple redémarre les instances EC2 dans l'unité `ou-1a2b3c-4d5e6c` AWS d'organisation. Les instances sont situées dans les régions `us-west-1` et `us-west-2`. Les instances doivent être membres du groupe de ressources AWS `WebServices`.

Linux

```
aws ssm start-automation-execution \
    --document-name AWS-RestartEC2Instance \
    --parameters AutomationAssumeRole=arn:aws:iam::123456789012:role/AWS-
SystemsManager-AutomationAdministrationRole \
    --target-parameter-name InstanceId \
    --targets Key=ResourceGroup,Values=WebServices \
    --target-locations Accounts=ou-1a2b3c-4d5e6c,Regions=us-west-1,us-
west-2,ExecutionRoleName=AWS-SystemsManager-AutomationExecutionRole
```

Windows

```
aws ssm start-automation-execution ^
    --document-name AWS-RestartEC2Instance ^
    --parameters AutomationAssumeRole=arn:aws:iam::123456789012:role/AWS-
SystemsManager-AutomationAdministrationRole ^
    --target-parameter-name InstanceId ^
    --targets Key=ResourceGroup,Values=WebServices ^
    --target-locations Accounts=ou-1a2b3c-4d5e6c,Regions=us-west-1,us-
west-2,ExecutionRoleName=AWS-SystemsManager-AutomationExecutionRole
```

PowerShell

```
$Targets = New-Object Amazon.SimpleSystemsManagement.Model.Target
$Targets.Key = "ResourceGroup"
$Targets.Values = "WebServices"

Start-SSMAutomationExecution `
    -DocumentName "AWS-RestartEC2Instance" `
    -Parameter @{
    "AutomationAssumeRole"="arn:aws:iam::123456789012:role/AWS-SystemsManager-
AutomationAdministrationRole" } `
    -TargetParameterName "InstanceId" `
    -Target $Targets `
    -TargetLocation @{
    "Accounts"="ou-1a2b3c-4d5e6c";
```

```
    "Regions"="us-west-1";
    "ExecutionRoleName"="AWS-SystemsManager-AutomationExecutionRole" }
```

Le système renvoie des informations similaires à ce qui suit :

Linux

```
{
    "AutomationExecutionId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE"
}
```

Windows

```
{
    "AutomationExecutionId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE"
}
```

PowerShell

```
4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE
```

3. Exécutez la commande suivante pour afficher les détails de l'automatisation.

   Linux

```
aws ssm describe-automation-executions \
    --filters Key=ExecutionId,Values=4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE
```

   Windows

```
aws ssm describe-automation-executions ^
    --filters Key=ExecutionId,Values=4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE
```

   PowerShell

```
Get-SSMAutomationExecutionList | `
    Where {$_.AutomationExecutionId -eq "a4a3c0e9-7efd-462a-8594-01234EXAMPLE"}
```

4. Exécutez la commande suivante pour afficher des détails sur la progression de l'automatisation.

   Linux

```
aws ssm get-automation-execution \
    --automation-execution-id 4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE
```

   Windows

```
aws ssm get-automation-execution ^
    --automation-execution-id 4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE
```

   PowerShell

```
Get-SSMAutomationExecution `
    -AutomationExecutionId a4a3c0e9-7efd-462a-8594-01234EXAMPLE
```

> **Note**
>
> Vous pouvez également surveiller le statut de l'automatisation dans la console. Dans la liste des exécutions d'Automation, choisissez l'automatisation que vous venez d'exécuter, puis choisissez l'onglet Execution steps (Étapes d'exécution). Cet onglet affiche le statut des actions d'automatisation.

Contenu connexe

Des correctifs centralisés sur plusieurs comptes et plusieurs régions avec AWS Systems Manager Automation

Référence sur les actions Systems Manager Automation

Cette référence décrit les actions Automation que vous pouvez spécifier dans un document AWS Systems Manager Automation. Ces actions ne peuvent pas être utilisées dans d'autres types de documents SSM. Pour de plus amples informations sur les plug-ins pour d'autres types de documents SSM, veuillez consulter Référence du plug-in de document Systems Manager (p. 1232).

Systems Manager Automation exécute les étapes décrites dans les documents Automation. Chaque étape est associée à une action spécifique. L'action détermine les entrées, le comportement et les sorties de l'étape. Les étapes sont définies dans la section `mainSteps` de votre document d'Automation.

Vous n'avez pas besoin de spécifier les sorties d'une action ou d'une étape. Les sorties sont prédéterminées par l'action associée à l'étape. Lorsque vous spécifiez des entrées d'étape dans vos documents Automation, vous pouvez référencer une ou plusieurs sorties d'une étape précédente. Par exemple, vous pouvez rendre la sortie d'`aws:runInstances` disponible pour une action `aws:runCommand` suivante. Vous pouvez également référencer des sorties d'étapes précédentes dans la section `Output` du document Automation.

> **Important**
>
> Si vous exécutez un flux de travail d'automatisation qui appelle d'autres services à l'aide d'un rôle de service AWS Identity and Access Management (IAM), le rôle de service doit être configuré avec l'autorisation d'appeler ces services. Cette exigence s'applique à tous les documents AWS Automation (documents `AWS-*`) comme les documents `AWS-ConfigureS3BucketLogging`, `AWS-CreateDynamoDBBackup` et `AWS-RestartEC2Instance`, pour n'en citer que quelques-uns. Cette exigence s'applique également à tous les documents Automation personnalisés que vous créez qui appellent d'autres services AWS à l'aide d'actions qui appellent d'autres services. Par exemple, si vous utilisez les actions `aws:executeAwsApi`, `aws:createStack` ou `aws:copyImage`, vous devez configurer le rôle de service avec l'autorisation d'appeler ces services. Vous pouvez activer des autorisations à d'autres services AWS en ajoutant une stratégie IAM en ligne au rôle. Pour de plus amples informations, veuillez consulter (Facultatif) Ajoutez une stratégie en ligne Automation pour appeler d'autres services AWS (p. 311).

Rubriques

- Propriétés partagées par toutes les actions (p. 387)
- aws:approve – Suspendre une exécution pour approbation manuelle (p. 392)
- aws:assertAwsResourceProperty – Affirmer un état de ressource AWS ou un état d'événement (p. 395)
- aws:branch – Exécuter les étapes d'automatisation conditionnelle (p. 397)

- aws:changeInstanceState – Modifier ou affirmer l'état de l'instance (p. 399)
- aws:copyImage – Copier ou chiffrer une Amazon Machine Image (p. 401)
- aws:createImage – Créer une Amazon Machine Image (p. 403)
- aws:createStack – Créer une pile AWS CloudFormation (p. 404)
- aws:createTags – Créer des balises pour les ressources AWS (p. 410)
- aws:deleteImage – Supprimer une Amazon Machine Image (p. 412)
- aws:deleteStack – Supprimer une pile AWS CloudFormation (p. 412)
- aws:executeAutomation – Exécuter une autre exécution d'automatisation (p. 414)
- aws:executeAwsApi – Appeler et exécuter des actions de l'API AWS (p. 416)
- aws:executeScript – Exécuter un script (p. 418)
- aws:executeStateMachine – Exécuter une machine d'état AWS Step Functions (p. 421)
- aws:invokeLambdaFunction – Appeler une fonction AWS Lambda (p. 422)
- aws:pause – Suspendre une exécution d'automatisation (p. 424)
- aws:runCommand – Exécuter une commande sur une instance gérée (p. 424)
- aws:runInstances – Lancer une instance EC2 (p. 429)
- aws:sleep – Retarder une exécution d'automatisation (p. 433)
- aws:waitForAwsResourceProperty – Attendre une propriété de ressource AWS (p. 435)
- Variables système Automation (p. 437)

Propriétés partagées par toutes les actions

Les propriétés communes sont des paramètres ou des options qui se trouvent dans toutes les actions. Certaines options définissent le comportement d'exécution d'une étape, par exemple le temps d'attente pour qu'une étape se termine et ce qu'il faut faire en cas d'échec de l'étape. Les propriétés suivantes sont communes à toutes les actions.

name (p. 389)

Identifiant qui doit être unique pour tous les noms d'étape dans le document.

Type: String (chaîne)

Obligatoire : Oui

action (p. 389)

Nom de l'action que l'étape doit exécuter. aws:runCommand – Exécuter une commande sur une instance gérée (p. 424) est un exemple d'action que vous pouvez spécifier ici. Ce document fournit des informations détaillées sur toutes les actions disponibles.

Type: String (chaîne)

Obligatoire : Oui

maxAttempts (p. 389)

Nombre de fois où l'étape doit être réessayée en cas de défaillance. Si la valeur est supérieure à 1, l'étape n'est pas considérée comme ayant échoué tant que toutes les tentatives n'ont pas échoué. La valeur par défaut est 1.

Type: Integer

Obligatoire : Non

timeoutSeconds (p. 389)

Valeur de l'expiration du délai d'exécution de l'étape. Si le délai est expiré et que la valeur de `maxAttempts` est supérieure à 1, alors l'étape n'est pas considérée comme ayant expiré tant que toutes les tentatives n'ont pas été effectuées.

Type: Integer

Obligatoire : Non

onFailure (p. 389)

Indique si le flux de travail doit être abandonné, poursuivi ou changer d'étape en cas d'échec. La valeur par défaut de cette option est abort.

Type: String (chaîne)

Valeurs valides : Abort | Continue | step:*`step_name`*

Obligatoire : Non

isEnd (p. 390)

Cette option arrête une exécution d'Automation à la fin d'une étape spécifique. L'exécution d'Automation s'arrête si l'exécution de l'étape a échoué ou réussi. La valeur par défaut est false.

Type: Booléen

Valeurs valides : true | false

Obligatoire : Non

nextStep (p. 390)

Spécifie l'étape du flux de travail d'automatisation à traiter immédiatement après la fin d'une étape.

Type: String (chaîne)

Obligatoire : Non

isCritical (p. 391)

Désigne une étape comme étant critique pour la réussite de l'exécution d'Automation. Si une étape portant cette désignation échoue, Automation signale l'état final de l'automatisation comme Échouée. Cette propriété est uniquement évaluée si vous la définissez explicitement dans votre étape. Si la `onFailure` propriété est définie sur `Continue` dans une étape, la valeur par défaut est false. Sinon, la valeur par défaut de cette option est true.

Type: Booléen

Valeurs valides : true | false

Obligatoire : Non

inputs (p. 391)

Propriétés spécifiques à l'action.

Type: Map

Obligatoire : Oui

Example

```
---
description: "Custom Automation Example"
schemaVersion: '0.3'
assumeRole: "{{ AutomationAssumeRole }}"
parameters:
  AutomationAssumeRole:
    type: String
    description: "(Required) The ARN of the role that allows Automation to perform
      the actions on your behalf. If no role is specified, Systems Manager Automation
      uses your IAM permissions to run this document."
    default: ''
  InstanceId:
      type: String
      description: "(Required) The Instance Id whose root EBS volume you want to restore
 the latest Snapshot."
      default: ''
mainSteps:
- name: getInstanceDetails
  action: aws:executeAwsApi
  onFailure: Abort
  inputs:
    Service: ec2
    Api: DescribeInstances
    InstanceIds:
    - "{{ InstanceId }}"
  outputs:
    - Name: availabilityZone
      Selector: "$.Reservations[0].Instances[0].Placement.AvailabilityZone"
      Type: String
    - Name: rootDeviceName
      Selector: "$.Reservations[0].Instances[0].RootDeviceName"
      Type: String
  nextStep: getRootVolumeId
- name: getRootVolumeId
  action: aws:executeAwsApi
  maxAttempts: 3
  onFailure: Abort
  inputs:
    Service: ec2
    Api: DescribeVolumes
    Filters:
    -  Name: attachment.device
       Values: ["{{ getInstanceDetails.rootDeviceName }}"]
    -  Name: attachment.instance-id
       Values: ["{{ InstanceId }}"]
  outputs:
    - Name: rootVolumeId
      Selector: "$.Volumes[0].VolumeId"
      Type: String
  nextStep: getSnapshotsByStartTime
- name: getSnapshotsByStartTime
  action: aws:executeScript
  timeoutSeconds: 45
  onFailure: Abort
  inputs:
    Runtime: python3.6
    Handler: getSnapshotsByStartTime
    InputPayload:
      rootVolumeId : "{{ getRootVolumeId.rootVolumeId }}"
    Script: |-
```

```
      def getSnapshotsByStartTime(events,context):
        import boto3

        #Initialize client
        ec2 = boto3.client('ec2')
        rootVolumeId = events['rootVolumeId']
        snapshotsQuery = ec2.describe_snapshots(
          Filters=[
            {
              "Name": "volume-id",
              "Values": [rootVolumeId]
            }
          ]
        )
        if not snapshotsQuery['Snapshots']:
          noSnapshotFoundString = "NoSnapshotFound"
          return { 'noSnapshotFound' : noSnapshotFoundString }
        else:
          jsonSnapshots = snapshotsQuery['Snapshots']
          sortedSnapshots = sorted(jsonSnapshots, key=lambda k: k['StartTime'],
 reverse=True)
          latestSortedSnapshotId = sortedSnapshots[0]['SnapshotId']
          return { 'latestSnapshotId' : latestSortedSnapshotId }
  outputs:
  - Name: Payload
    Selector: $.Payload
    Type: StringMap
  - Name: latestSnapshotId
    Selector: $.Payload.latestSnapshotId
    Type: String
  - Name: noSnapshotFound
    Selector: $.Payload.noSnapshotFound
    Type: String
  nextStep: branchFromResults
- name: branchFromResults
  action: aws:branch
  onFailure: Abort
  inputs:
    Choices:
    - NextStep: createNewRootVolumeFromSnapshot
      Not:
        Variable: "{{ getSnapshotsByStartTime.noSnapshotFound }}"
        StringEquals: "NoSnapshotFound"
  isEnd: true
- name: createNewRootVolumeFromSnapshot
  action: aws:executeAwsApi
  onFailure: Abort
  inputs:
    Service: ec2
    Api: CreateVolume
    AvailabilityZone: "{{ getInstanceDetails.availabilityZone }}"
    SnapshotId: "{{ getSnapshotsByStartTime.latestSnapshotId }}"
  outputs:
    - Name: newRootVolumeId
      Selector: "$.VolumeId"
      Type: String
  nextStep: stopInstance
- name: stopInstance
  action: aws:executeAwsApi
  onFailure: Abort
  inputs:
    Service: ec2
    Api: StopInstances
    InstanceIds:
    - "{{ InstanceId }}"
  nextStep: verifyVolumeAvailability
```

```
- name: verifyVolumeAvailability
  action: aws:waitForAwsResourceProperty
  timeoutSeconds: 120
  inputs:
    Service: ec2
    Api: DescribeVolumes
    VolumeIds:
    - "{{ createNewRootVolumeFromSnapshot.newRootVolumeId }}"
    PropertySelector: "$.Volumes[0].State"
    DesiredValues:
    - "available"
  nextStep: verifyInstanceStopped
- name: verifyInstanceStopped
  action: aws:waitForAwsResourceProperty
  timeoutSeconds: 120
  inputs:
    Service: ec2
    Api: DescribeInstances
    InstanceIds:
    - "{{ InstanceId }}"
    PropertySelector: "$.Reservations[0].Instances[0].State.Name"
    DesiredValues:
    - "stopped"
  nextStep: detachRootVolume
- name: detachRootVolume
  action: aws:executeAwsApi
  onFailure: Abort
  isCritical: true
  inputs:
    Service: ec2
    Api: DetachVolume
    VolumeId: "{{ getRootVolumeId.rootVolumeId }}"
  nextStep: verifyRootVolumeDetached
- name: verifyRootVolumeDetached
  action: aws:waitForAwsResourceProperty
  timeoutSeconds: 30
  inputs:
    Service: ec2
    Api: DescribeVolumes
    VolumeIds:
    - "{{ getRootVolumeId.rootVolumeId }}"
    PropertySelector: "$.Volumes[0].State"
    DesiredValues:
    - "available"
  nextStep: attachNewRootVolume
- name: attachNewRootVolume
  action: aws:executeAwsApi
  onFailure: Abort
  inputs:
    Service: ec2
    Api: AttachVolume
    Device: "{{ getInstanceDetails.rootDeviceName }}"
    InstanceId: "{{ InstanceId }}"
    VolumeId: "{{ createNewRootVolumeFromSnapshot.newRootVolumeId }}"
  nextStep: verifyNewRootVolumeAttached
- name: verifyNewRootVolumeAttached
  action: aws:waitForAwsResourceProperty
  timeoutSeconds: 30
  inputs:
    Service: ec2
    Api: DescribeVolumes
    VolumeIds:
    - "{{ createNewRootVolumeFromSnapshot.newRootVolumeId }}"
    PropertySelector: "$.Volumes[0].Attachments[0].State"
    DesiredValues:
    - "attached"
```

```
  nextStep: startInstance
- name: startInstance
  action: aws:executeAwsApi
  onFailure: Abort
  inputs:
    Service: ec2
    Api: StartInstances
    InstanceIds:
    - "{{ InstanceId }}"
```

aws:approve – Suspendre une exécution pour approbation manuelle

Interrompt temporairement une exécution d'Automation jusqu'à ce que les mandataires désignés aient approuvé ou rejeté l'action. Une fois le nombre d'approbations requises atteint, l'exécution d'Automation reprend. Vous pouvez insérer l'étape d'approbation n'importe quel emplacement dans la mainSteps section de votre document d'Automation.

> **Note**
>
> Le délai d'expiration par défaut est de 7 jours pour cette action (604 800 secondes). Vous pouvez limiter ou étendre le délai d'attente en spécifiant le paramètre `timeoutSeconds` pour une action aws:approve step. Si l'étape Automation atteint la valeur de délai d'attente avant de recevoir toutes les approbations requises, alors l'étape et l'instance d'Automation s'arrêtent et renvoient le statut Expiré.

Dans l'exemple suivant, l'action aws;approve interrompt temporairement le flux de travail d'automatisation jusqu'à ce qu'un approbateur accepte ou rejette le flux de travail. Après approbation, le document exécute une PowerShell commande simple.

YAML

```
---
description: RunInstancesDemo1
schemaVersion: '0.3'
assumeRole: "{{ assumeRole }}"
parameters:
  assumeRole:
    type: String
  message:
    type: String
mainSteps:
- name: approve
  action: aws:approve
  timeoutSeconds: 1000
  onFailure: Abort
  inputs:
    NotificationArn: arn:aws:sns:us-east-2:12345678901:AutomationApproval
    Message: "{{ message }}"
    MinRequiredApprovals: 1
    Approvers:
    - arn:aws:iam::12345678901:user/AWS-User-1
- name: run
  action: aws:runCommand
  inputs:
    InstanceIds:
    - i-1a2b3c4d5e6f7g
    DocumentName: AWS-RunPowerShellScript
    Parameters:
      commands:
      - date
```

JSON

```
{
   "description":"RunInstancesDemo1",
   "schemaVersion":"0.3",
   "assumeRole":"{{ assumeRole }}",
   "parameters":{
      "assumeRole":{
         "type":"String"
      },
      "message":{
         "type":"String"
      }
   },
   "mainSteps":[
      {
         "name":"approve",
         "action":"aws:approve",
         "timeoutSeconds":1000,
         "onFailure":"Abort",
         "inputs":{
            "NotificationArn":"arn:aws:sns:us-east-2:12345678901:AutomationApproval",
            "Message":"{{ message }}",
            "MinRequiredApprovals":1,
            "Approvers":[
               "arn:aws:iam::12345678901:user/AWS-User-1"
            ]
         }
      },
      {
         "name":"run",
         "action":"aws:runCommand",
         "inputs":{
            "InstanceIds":[
               "i-1a2b3c4d5e6f7g"
            ],
            "DocumentName":"AWS-RunPowerShellScript",
            "Parameters":{
               "commands":[
                  "date"
               ]
            }
         }
      }
   ]
}
```

Vous pouvez approuver ou refuser des automatisations en attente d'approbation dans la console.

Pour approuver ou rejeter des automatisations en attente

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Automatisation.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Automation.
3. Choisissez l'option en regard d'un flux de travail Automation avec le statut En attente.

4. Choisissez Approve/Deny (Approuver/Refuser).
5. Vérifiez les détails de l'automatisation.
6. Sélectionnez Approuver ou Refuser, saisissez un commentaire facultatif, puis choisissez Soumettre.

Entrées

YAML

```
NotificationArn: arn:aws:sns:us-west-1:12345678901:Automation-ApprovalRequest
Message: Please approve this step of the Automation.
MinRequiredApprovals: 3
Approvers:
- IamUser1
- IamUser2
- arn:aws:iam::12345678901:user/IamUser3
- arn:aws:iam::12345678901:role/IamRole
```

JSON

```
{
   "NotificationArn":"arn:aws:sns:us-west-1:12345678901:Automation-ApprovalRequest",
   "Message":"Please approve this step of the Automation.",
   "MinRequiredApprovals":3,
   "Approvers":[
      "IamUser1",
      "IamUser2",
      "arn:aws:iam::12345678901:user/IamUser3",
      "arn:aws:iam::12345678901:role/IamRole"
   ]
}
```

NotificationArn

L'ARN d'une rubrique Amazon SNS pour les approbations d'automatisation. Lorsque vous spécifiez une étape `aws:approve` dans un document Automation, Automation envoie un message à cette rubrique, permettant aux mandataires de savoir qu'ils doivent approuver ou rejeter une étape d'automatisation. Le titre de la rubrique Amazon SNS doit avoir un préfixe contenant « Automation ».

Type: String (chaîne)

Obligatoire : Non

Message

Les informations que vous souhaitez inclure dans la rubrique SNS lorsque la requête d'approbation est envoyée. Le message peut contenir un nombre maximum de 4 096 caractères.

Type: String (chaîne)

Obligatoire : Non

MinRequiredApprovals

Le nombre minimum d'approbations requises pour que l'exécution d'Automation reprenne. Si vous ne spécifiez aucune valeur, le système en établit une par défaut. La valeur de ce paramètre doit être un nombre positif. La valeur de ce paramètre ne peut pas dépasser le nombre d'approbateurs défini par le paramètre `Approvers`.

Type: Integer

Obligatoire : Non

Approbateurs

Une liste des mandataires AWS authentifiés qui ont la capacité d'approuver ou de rejeter l'action. Le nombre maximum d'approbateurs est de 10. Vous pouvez spécifier des mandataires à l'aide des formats suivants :

- Nom d'utilisateur AWS Identity and Access Management (IAM)
- Un ARN d'utilisateur IAM
- Un ARN de rôle IAM
- Un ARN d'utilisateur responsable IAM

Type: StringList

Obligatoire : Oui

Output

ApprovalStatus

L'état d'approbation de l'étape. Le statut peut avoir l'une des valeurs suivantes : en attente, rejeté ou approuvé. En attente signifie qu'Automation attend la saisie des approbateurs.

Type: String (chaîne)

ApproverDecisions

Une carte JSON incluant la décision d'approbation de chaque approbateur.

Type: MapList

aws:assertAwsResourceProperty – Affirmer un état de ressource AWS ou un état d'événement

L'action aws:assertAwsResourceProperty vous permet d'imposer un état de ressource ou d'événement spécifique pour une étape d'automatisation. Par exemple, vous pouvez spécifier qu'une étape d'automatisation doit attendre le démarrage d'une instance EC2. Elle appellera ensuite l'action Amazon EC2 DescribeInstanceStatus d'API avec la DesiredValue propriété de `running` . Cela garantit que le flux de travail Automation attend une instance en cours d'exécution, puis continue lorsque l'instance est en cours d'exécution.

Pour de plus amples informations et des exemples sur l'utilisation de cette action, veuillez consulter Appel d'autres services AWS à partir d'un flux de travail Systems Manager Automation (p. 470).

Note

La valeur de délai d'expiration par défaut pour cette action est de 3600 secondes (une heure). Vous pouvez limiter ou étendre le délai d'attente en spécifiant le paramètre `timeoutSeconds` pour une étape `aws:assertAwsResourceProperty`.

Input

Les entrées sont définies par l'action d'API que vous choisissez.

YAML

```
action: aws:assertAwsResourceProperty
inputs:
  Service: The official namespace of the service
  Api: The API action or method name
  API action inputs or parameters: A value
  PropertySelector: Response object
  DesiredValues:
  - Desired property values
```

JSON

```
{
  "action": "aws:assertAwsResourceProperty",
  "inputs": {
    "Service":"The official namespace of the service",
    "Api":"The API action or method name",
    "API action inputs or parameters":"A value",
    "PropertySelector": "Response object",
    "DesiredValues": [
      "Desired property values"
    ]
  }
}
```

Service

L'espace de noms de service AWS qui contient l'action d'API que vous souhaitez exécuter. Par exemple, l'espace de noms pour Systems Manager est `ssm` . L'espace de noms pour Amazon EC2 est `ec2` . Vous pouvez afficher une liste des espaces de noms de service AWS pris en charge dans la section Services disponibles de .AWS CLI Command Reference

Type: String (chaîne)

Obligatoire : Oui

Api

Le nom de l'action d'API que vous voulez exécuter. Vous pouvez afficher les actions d'API (également appelées méthodes) en choisissant un service dans le panneau de navigation de gauche sur la page Services Reference (Référence des services) suivante. Choisissez une méthode dans la section Client pour le service que vous voulez appeler. Par exemple, toutes les actions d'API (méthodes) pour Amazon Relational Database Service (Amazon RDS) sont répertoriées sur la page suivante : Méthodes Amazon RDS.

Type: String (chaîne)

Obligatoire : Oui

Entrées d'action d'API

Une ou plusieurs entrées d'action d'API. Vous pouvez afficher les entrées disponibles (également appelées paramètres) en choisissant un service dans le panneau de navigation de gauche sur la page Services Reference (Référence des services) suivante. Choisissez une méthode dans la section Client pour le service que vous voulez appeler. Par exemple, toutes les méthodes pour Amazon RDS sont répertoriées sur la page suivante : Méthodes Amazon RDS. Choisissez la méthode

describe_db_instances et faites défiler vers le bas pour voir les paramètres disponibles, tels que DBInstanceIdentifier , Name et Values. Utilisez le format suivant pour spécifier plusieurs entrées.

YAML

```
inputs:
  Service: The official namespace of the service
  Api: The API action name
  API input 1: A value
  API Input 2: A value
  API Input 3: A value
```

JSON

```
"inputs":{
      "Service":"The official namespace of the service",
      "Api":"The API action name",
      "API input 1":"A value",
      "API Input 2":"A value",
      "API Input 3":"A value"
}
```

Type: Déterminé par l'action d'API choisie

Obligatoire : Oui

PropertySelector

L'attribut JSONPath spécifique dans l'objet de réponse. Vous pouvez afficher les objets de réponse en choisissant un service dans le panneau de navigation de gauche sur la page Services Reference (Référence des services) suivante. Choisissez une méthode dans la section Client pour le service que vous voulez appeler. Par exemple, toutes les méthodes pour Amazon RDS sont répertoriées sur la page suivante : Méthodes Amazon RDS. Choisissez la méthode describe_db_instances et faites défiler la page jusqu'à la section Response Structure (Structure d'une réponse). DBInstancesest répertorié en tant qu'objet de réponse.

Type: Entier, Booléen, Chaîne, StringList , StringMap , ouMapList

Obligatoire : Oui

DesiredValues

État attendu ou état à partir duquel le flux de travail d'automatisation se poursuit. Si vous spécifiez une valeur booléenne, vous devez utiliser une majuscule, comme True ou False.

Type: Varie

Obligatoire : Oui

aws:branch – Exécuter les étapes d'automatisation conditionnelle

L'action `aws:branch` vous permet de créer un flux de travail d'automatisation dynamique qui évalue différentes options en une seule étape, puis passe à une autre étape dans le document d'Automation en fonction des résultats de cette évaluation.

Lorsque vous spécifiez l'action `aws:branch` pour une étape, vous définissez des `Choices` que le flux de travail doit évaluer. Les `Choices` peuvent se baser sur une valeur que vous avez spécifiée dans la section `Parameters` du document d'Automation ou sur une valeur dynamique générée comme sortie à partir de l'étape précédente. Le flux de travail d'automatisation évalue chaque choix à l'aide d'une expression booléenne. Si le premier choix est true, le flux de travail passe à l'étape désignée pour ce choix. Si le premier choix est false, le flux de travail évalue le choix suivant. Le flux de travail continue d'évaluer chaque

choix jusqu'à ce qu'il traite un choix défini sur true. Le flux de travail accède ensuite à l'étape désignée correspondant au choix défini sur true.

Si aucun des choix n'est true, le flux de travail vérifie si l'étape contient une valeur `default`. La valeur par défaut définit une étape à laquelle le flux de travail doit passer si aucun des choix n'est défini sur true. Si aucune valeur `default` n'est spécifiée pour l'étape, le flux de travail d'automatisation traite l'étape suivante du document.

L'action `aws:branch` prend en charge les évaluations de choix complexes à l'aide d'une combinaison d'opérateurs `And`, `Not` et `Or`. Pour de plus amples informations sur l'utilisation d'`aws:branch`, y compris des exemples de documents et des exemples utilisant différents opérateurs, veuillez consulter Création de flux de travail Automation dynamiques avec branchement conditionnel (p. 459).

Input

Spécifiez un ou plusieurs `Choices` dans une étape. Les `Choices` peuvent se baser sur une valeur que vous avez spécifiée dans la section `Parameters` du document d'Automation ou sur une valeur dynamique générée comme sortie à partir de l'étape précédente. Voici un exemple de fichier YAML qui évalue un paramètre.

```
mainSteps:
- name: chooseOS
  action: aws:branch
  inputs:
    Choices:
    - NextStep: runWindowsCommand
      Variable: "{{Name of a parameter defined in the Parameters section. For example:
 OS_name}}"
      StringEquals: windows
    - NextStep: runLinuxCommand
      Variable: "{{Name of a parameter defined in the Parameters section. For example:
 OS_name}}"
      StringEquals: linux
    Default:
      sleep3
```

Voici un exemple de fichier YAML qui évalue la sortie d'une étape précédente.

```
mainSteps:
- name: chooseOS
  action: aws:branch
  inputs:
    Choices:
    - NextStep: runPowerShellCommand
      Variable: "{{Name of a response object. For example: GetInstance.platform}}"
      StringEquals: Windows
    - NextStep: runShellCommand
      Variable: "{{Name of a response object. For example: GetInstance.platform}}"
      StringEquals: Linux
    Default:
      sleep3
```

Choix

Une ou plusieurs expressions qu'Automation doit évaluer pour déterminer l'étape suivante à traiter. L'évaluation des choix repose sur une expression booléenne. Chaque choix doit définir les options suivantes :

- NextStep: Étape suivante à traiter dans le document Automation si le choix désigné est true.
- Variable : Spécifiez le nom d'un paramètre qui est défini dans la `Parameters` section du document d'Automation. Vous pouvez également spécifier un objet de sortie d'une étape précédente dans

le document d'Automation. Pour de plus amples informations sur la création de variables pour `aws:branch`, veuillez consulter À propos de la création de la variable de sortie (p. 463).

- Opération : Critères utilisés pour évaluer le choix. L'action `aws:branch` prend en charge les opérations suivantes :

  Opérations de chaîne

  - StringEquals
  - EqualsIgnoreCase
  - StartsWith
  - EndsWith
  - Contains

  Opérations numériques

  - NumericEquals
  - NumericGreater
  - NumericLesser
  - NumericGreaterOrEquals
  - NumericLesser
  - NumericLesserOrEquals

  Opération booléenne

  - BooleanEquals

    Important

    Lorsque vous créez un document d'Automation, le système vérifie chaque opération dans le document. Si une opération n'est pas prise en charge, le système renvoie une erreur lorsque vous tentez de créer le document.

Valeur par défaut

Nom d'une étape à laquelle le flux de travail doit passer si aucun des `Choices` n'est true.

Type: String (chaîne)

Obligatoire : Non

Note

`Laws:branch`'action prend en charge les `And` opérateurs `Or` , et .`Not` Pour obtenir des exemples `aws:branch` qui utilisent des opérateurs, veuillez consulter Création de flux de travail Automation dynamiques avec branchement conditionnel (p. 459).

aws:changeInstanceState – Modifier ou affirmer l'état de l'instance

Modifie ou affirme l'état de l'instance.

Cette action peut être utilisée en mode déclaration (n'exécutez pas l'API pour modifier l'état, mais vérifiez que l'instance est à l'état souhaité.) Pour utiliser le mode assert, définissez le `CheckStateOnly` paramètre sur true. Ce mode est utile lorsque vous exécutez la commande Sysprep sous Windows, une commande asynchrone qui peut être exécutée en arrière-plan pendant longtemps. Vous pouvez veiller à ce que l'instance soit arrêtée avant de créer une AMI.

Note

La valeur de délai d'expiration par défaut pour cette action est de 3600 secondes (une heure). Vous pouvez limiter ou étendre le délai d'attente en spécifiant le paramètre `timeoutSeconds` pour une étape `aws:changeInstanceState`.

Entrées

YAML

```
name: stopMyInstance
action: aws:changeInstanceState
maxAttempts: 3
timeoutSeconds: 3600
onFailure: Abort
inputs:
  InstanceIds:
  - i-1234567890abcdef0
  CheckStateOnly: true
  DesiredState: stopped
```

JSON

```
{
    "name":"stopMyInstance",
    "action": "aws:changeInstanceState",
    "maxAttempts": 3,
    "timeoutSeconds": 3600,
    "onFailure": "Abort",
    "inputs": {
        "InstanceIds": ["i-1234567890abcdef0"],
        "CheckStateOnly": true,
        "DesiredState": "stopped"
    }
}
```

InstanceIds

de ces instances.IDs

Type: StringList

Obligatoire : Oui

CheckStateOnly

Si la valeur est false, définit l'état de l'instance sur l'état souhaité. Si la valeur est true, déclare l'état souhaité à l'aide de l'interrogation.

Par défaut : `false`

Type: Booléen

Obligatoire : Non

DesiredState

État souhaité. Lorsqu'elle est définie sur `running`, cette action attend que l'état Amazon EC2 indique `Running`, que le statut d'instance soit `OK` et que le statut système corresponde à `OK` avant de se terminer.

Type: String (chaîne)

Valeurs valides : `running| | |stoppedterminated`

Obligatoire : Oui

Force

Si ce paramètre est défini, il oblige les instances à s'arrêter. Les instances n'ont pas la possibilité de vider les caches du système de fichiers ou les métadonnées du système de fichiers. Si vous utilisez cette option, vous devez effectuer un contrôle du système de fichiers et des procédures de réparation. Cette option n'est pas recommandée pour les instances EC2 pour Windows Server.

Type: Booléen

Obligatoire : Non

AdditionalInfo

Instances réservées.

Type: String (chaîne)

Obligatoire : Non

Output

None (Aucun)

aws:copyImage – Copier ou chiffrer une Amazon Machine Image

Copie une AMI de n'importe quelle région vers la région actuelle. Cette action peut également chiffrer la nouvelle AMI.

Input

Cette action prend en charge la plupart des CopyImage paramètres. Pour plus d'informations, consultez CopyImage .

L'exemple suivant crée une copie d'une AMI dans la région de Séoul (`SourceImageID`: ami-0fe10819. `SourceRegion`: ap-northeast-2). La nouvelle AMI est copiée dans la région dans laquelle vous avez lancé l'action d'automatisation. L'AMI copiée sera chiffrée, car l'indicateur `Encrypted` facultatif est défini sur `true`.

YAML

```
name: createEncryptedCopy
action: aws:copyImage
maxAttempts: 3
onFailure: Abort
inputs:
  SourceImageId: ami-0fe10819
  SourceRegion: ap-northeast-2
  ImageName: Encrypted Copy of LAMP base AMI in ap-northeast-2
  Encrypted: true
```

JSON

```
{
    "name": "createEncryptedCopy",
    "action": "aws:copyImage",
    "maxAttempts": 3,
    "onFailure": "Abort",
```

```
    "inputs": {
        "SourceImageId": "ami-0fe10819",
        "SourceRegion": "ap-northeast-2",
        "ImageName": "Encrypted Copy of LAMP base AMI in ap-northeast-2",
        "Encrypted": true
    }
}
```

SourceRegion

Région dans laquelle l'AMI source existe actuellement.

Type: String (chaîne)

Obligatoire : Oui

SourceImageId

ID d'AMI à copier à partir de la région source.

Type: String (chaîne)

Obligatoire : Oui

ImageName

Nom de la nouvelle image.

Type: String (chaîne)

Obligatoire : Oui

ImageDescription

Description de l'image cible.

Type: String (chaîne)

Obligatoire : Non

Encrypted

Chiffrement de l'AMI cible.

Type: Booléen

Obligatoire : Non

KmsKeyId

Amazon Resource Name (ARN) complet de la clé principale client (CMK) AWS Key Management Service à utiliser lors du chiffrement des instantanés d'une image pendant une opération de copie. Pour plus d'informations, consultez CopyImage .

Type: String (chaîne)

Obligatoire : Non

ClientToken

Identifiant unique, sensible à la casse, que vous devez fournir afin de garantir l'idempotence de la demande. Pour plus d'informations, consultez CopyImage .

Type: String (chaîne)

Obligatoire : Non

Output

ImageId

ID de l'image copiée.

ImageState

État de l'image copiée.

Valeurs valides : `available| | |pendingfailed`

aws:createImage – Créer une Amazon Machine Image

Crée une AMI à partir d'une instance en cours d'exécution ou arrêtée.

Input

Cette action prend en charge la plupart des CreateImage paramètres. Pour plus d'informations, consultez CreateImage .

YAML

```
name: createMyImage
action: aws:createImage
maxAttempts: 3
onFailure: Abort
inputs:
  InstanceId: i-1234567890abcdef0
  ImageName: AMI Created on{{global:DATE_TIME}}
  NoReboot: true
  ImageDescription: My newly created AMI
```

JSON

```
{
    "name": "createMyImage",
    "action": "aws:createImage",
    "maxAttempts": 3,
    "onFailure": "Abort",
    "inputs": {
        "InstanceId": "i-1234567890abcdef0",
        "ImageName": "AMI Created on{{global:DATE_TIME}}",
        "NoReboot": true,
        "ImageDescription": "My newly created AMI"
    }
}
```

InstanceId

ID de l'instance.

Type: String (chaîne)

Obligatoire : Oui

ImageName

Nom de l'image .

Type: String (chaîne)

Obligatoire : Oui

ImageDescription

Description de l'image.

Type: String (chaîne)

Obligatoire : Non

NoReboot

Littéral booléen.

Par défaut, Amazon EC2 tente d'arrêter et redémarrer l'instance avant de créer l'image. Si l'option Pas de redémarrage est définie sur `true`, Amazon EC2 n'arrête pas l'instance avant de créer l'image. Une fois cette option utilisée, l'intégrité du système de fichiers sur l'image créée ne peut pas être garantie.

Si vous ne souhaitez pas que l'instance s'exécute une fois que vous avez créé une image AMI à partir de celle-ci, utilisez d'abord laws:changeInstanceState – Modifier ou affirmer l'état de l'instance (p. 399)'action pour arrêter l'instance, puis utilisez cette `aws:createImage` action avec l'NoRebootoption définie sur `true` .

Type: Booléen

Obligatoire : Non

BlockDeviceMappings

Périphériques de stockage en mode bloc pour l'instance.

Type: Map

Obligatoire : Non

Output

ImageId

ID de l'image nouvellement créée.

ImageState

État actuel de l'image. Si l'état est disponible, l'image est enregistrée avec succès et peut être utilisée pour lancer une instance.

aws:createStack – Créer une pile AWS CloudFormation

Crée une nouvelle pile AWS CloudFormation à partir d'un modèle.

Pour plus d'informations sur la création de AWS CloudFormation piles, consultez CreateStack dans le AWS CloudFormation API Reference .

Entrées

YAML

```
name: makeStack
action: aws:createStack
maxAttempts: 1
onFailure: Abort
```

```
inputs:
  Capabilities:
  - CAPABILITY_IAM
  StackName: myStack
  TemplateURL: http://s3.amazonaws.com/doc-example-bucket/myStackTemplate
  TimeoutInMinutes: 5
```

JSON

```
{
    "name": "makeStack",
    "action": "aws:createStack",
    "maxAttempts": 1,
    "onFailure": "Abort",
    "inputs": {
        "Capabilities": [
            "CAPABILITY_IAM"
        ],
        "StackName": "myStack",
        "TemplateURL": "http://s3.amazonaws.com/doc-example-bucket/myStackTemplate",
        "TimeoutInMinutes": 5
    }
}
```

Capacités

Une liste de valeurs que vous spécifiez avant que AWS CloudFormation ne puisse créer certaines piles. Certains modèles de piles incluent des ressources pouvant affecter les autorisations de votre compte AWS. Par exemple, la création d'utilisateurs AWS Identity and Access Management (IAM) peut affecter les autorisations de votre compte. Pour ces piles, vous devez explicitement reconnaître leurs capacités en spécifiant ce paramètre.

Les valeurs valides sont `CAPABILITY_IAM`, `CAPABILITY_NAMED_IAM` et `CAPABILITY_AUTO_EXPAND`.

CAPABILITY_IAM et CAPABILITY_NAMED_IAM

Si vous disposez de ressources IAM, vous pouvez spécifier l'une ou l'autre de ces fonctionnalités. Si vous avez IAM des ressources avec des noms personnalisés, vous devez spécifier `CAPABILITY_NAMED_IAM` . Si vous ne spécifiez pas ce paramètre, cette action renvoie une `InsufficientCapabilities` erreur. Les ressources suivantes exigent que vous spécifiiez `CAPABILITY_IAM` ou `CAPABILITY_NAMED_IAM`.

- AWS::IAM::AccessKey
- AWS::IAM::Group
- AWS::IAM::InstanceProfile
- AWS::IAM::Policy
- AWS::IAM::Role
- AWS::IAM::User
- AWS::IAM::UserToGroupAddition

Si votre modèle de pile contient ces ressources, nous vous recommandons de vérifier toutes les autorisations qui y sont associées et de les modifier, si nécessaire.

Pour de plus amples informations, veuillez consulter Confirmation des ressources IAM dans les modèles AWS CloudFormation.

CAPABILITY_AUTO_EXPAND

Certains modèles contiennent des macros. Les macros effectuent un traitement personnalisé sur les modèles, pouvant inclure des actions simples comme des opérations de recherche et de remplacement, mais aussi des transformations d'envergure de modèles entiers. De ce fait, les utilisateurs créent généralement un jeu de modifications à partir du modèle traité, afin de pouvoir examiner les modifications résultant des macros avant de créer réellement la pile. Si votre modèle de pile contient une ou plusieurs macros et que vous choisissez de créer une pile directement à partir du modèle traité, sans examiner au préalable les modifications qui en résultent dans un jeu de modifications, vous devez reconnaître cette fonctionnalité.

Pour plus d'informations, consultez Utilisation des macros AWS CloudFormation pour effectuer un traitement personnalisé sur des modèles dans le AWS CloudFormation Guide de l'utilisateur.

Type : tableau de chaînes

Valeurs valides: `CAPABILITY_IAM` | `CAPABILITY_NAMED_IAM` | `CAPABILITY_AUTO_EXPAND`

Obligatoire : Non

ClientRequestToken

Identifiant unique de cette règle CreateStack. Spécifiez ce jeton si vous définissez maxAttempts dans cette étape une valeur supérieure à 1. En spécifiant ce jeton, vous informez AWS CloudFormation que vous ne tentez pas de créer une nouvelle pile avec le même nom.

Type: String (chaîne)

Obligatoire : Non

Contraintes de longueur : Longueur minimale de 1. Longueur maximale de 128.

Modèle : [a-zA-Z0-9][-a-zA-Z0-9]*

DisableRollback

Définir sur `true` pour désactiver la restauration de la pile si la création de pile a échoué.

Conditionnel : Vous pouvez spécifier le `DisableRollback` paramètre ou le `OnFailure` paramètre, mais pas les deux.

Par défaut : `false`

Type: Booléen

Obligatoire : Non

NotificationARNs

RubAmazon SNSrique ARNs de publication des événements liés à la pile. Vous pouvez trouver la rubrique SNS ARNs à l'aide de la Amazon SNS console https://console.aws.amazon.com/sns/v3/home , .

Type : tableau de chaînes

Membres du tableau : 5 éléments maximum.

Obligatoire : Non

OnFailure

Détermine l'action à mener si la création de pile a échoué. Vous devez spécifier `DO_NOTHING`, `ROLLBACK` ou `DELETE`.

Conditionnel : Vous pouvez spécifier le `OnFailure` paramètre ou le `DisableRollback` paramètre, mais pas les deux.

Par défaut : `ROLLBACK`

Type: String (chaîne)

Valeurs valides: `DO_NOTHING` | `ROLLBACK` | `DELETE`

Obligatoire : Non

Parameters

Une liste de structures `Parameter` qui spécifie les paramètres d'entrée pour la pile. Pour plus d'informations, consultez le type de données Paramètre.

Type : tableau d'objets Paramètre

Obligatoire : Non

ResourceTypes

Les types de ressource du modèle avec lesquels vous avez l'autorisation de travailler pour cette action de création de pile. Exemple : `AWS::EC2::Instance`, ou `AWS::EC2::*` .`Custom::MyCustomInstance` Utilisez la syntaxe suivante pour décrire les types de ressources du modèle.

- Pour toutes les ressources AWS :

```
AWS::*
```

- Pour toutes les ressources personnalisées :

```
Custom::*
```

- Pour une ressource personnalisée spécifique :

```
Custom::logical_ID
```

- Pour toutes les ressources d'un service AWS spécifique :

```
AWS::service_name::*
```

- Pour une ressource AWS spécifique :

```
AWS::service_name::resource_logical_ID
```

Si la liste de types de ressources n'inclut pas la ressource que vous êtes en train de créer, la création de pile échoue. Par défaut, AWS CloudFormation accorde des autorisations à tous les types de ressource. IAM utilise ce paramètre pour les clés de condition spécifiques à AWS CloudFormation dans les stratégies IAM. Pour plus d'informations, consultez Contrôle de l'accès avec AWS Identity and Access Management.

Type : tableau de chaînes

Contraintes de longueur : Longueur minimale de 1. Longueur maximale de 256.

Obligatoire : Non

RoleARN

Amazon Resource Name (ARN) d'un rôle IAM endossé par AWS CloudFormation pour créer la pile. AWS CloudFormation utilise les informations d'identification du rôle pour effectuer des appels en votre nom. AWS CloudFormation utilise toujours ce rôle pour toutes les opérations futures au niveau de la

pile. Tant que les utilisateurs ont l'autorisation d'opérer sur la pile, AWS CloudFormation utilise ce rôle même si les utilisateurs n'ont pas l'autorisation de le transmettre. Vérifiez que le rôle accorde le plus faible nombre de privilèges.

Si vous ne spécifiez pas de valeur, AWS CloudFormation utilise le rôle qui était précédemment associé à la pile. Si aucun rôle n'est disponible, AWS CloudFormation utilise une session temporaire qui est générée à partir de vos informations d'identification utilisateur.

Type: String (chaîne)

Contraintes de longueur : Longueur minimale de 20. Longueur maximale de 2048.

Obligatoire : Non

StackName

Le nom associé à la pile. Le nom doit être unique dans la région dans laquelle vous créez la pile.

> **Note**
>
> Un nom de pile 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 128 caractères.

Type: String (chaîne)

Obligatoire : Oui

StackPolicyBody

Structure contenant le corps de la stratégie de pile. Pour plus d'informations, consultez Empêchement des mises à jour des ressources de la pile.

Conditionnel : Vous pouvez spécifier le `StackPolicyBody` paramètre ou le `StackPolicyURL` paramètre, mais pas les deux.

Type: String (chaîne)

Contraintes de longueur : Longueur minimale de 1. Longueur maximale de 16384.

Obligatoire : Non

StackPolicyURL

Emplacement d'un fichier contenant la stratégie de pile. L'URL doit pointer vers une stratégie située dans un compartiment S3 de la même région que la pile. La taille maximum autorisée pour la stratégie de pile est 16 Ko.

Conditionnel : Vous pouvez spécifier le `StackPolicyBody` paramètre ou le `StackPolicyURL` paramètre, mais pas les deux.

Type: String (chaîne)

Contraintes de longueur : Longueur minimale de 1. Longueur maximale de 1350.

Obligatoire : Non

Tags

Paires clé-valeur à associer à cette pile. AWS CloudFormation propage aussi ces balises aux ressources créées dans la pile. Vous pouvez spécifier un nombre maximum de 10 balises.

Type : tableau d'objets Balise

Obligatoire : Non

TemplateBody

Structure contenant le corps du modèle avec une longueur minimale de 1 octet et une longueur maximale de 51 200 octets. Pour plus d'informations, consultez Anatomie du modèle.

Conditionnel : Vous pouvez spécifier le `TemplateBody` paramètre ou le `TemplateURL` paramètre, mais pas les deux.

Type: String (chaîne)

Contraintes de longueur : Longueur minimale de 1.

Obligatoire : Non

TemplateURL

Emplacement d'un fichier contenant le corps du modèle. L'URL doit pointer vers un modèle situé dans un compartiment S3. La taille maximum autorisée pour le modèle est 460 800 Ko. Pour plus d'informations, consultez Anatomie du modèle.

Conditionnel : Vous pouvez spécifier le `TemplateBody` paramètre ou le `TemplateURL` paramètre, mais pas les deux.

Type: String (chaîne)

Contraintes de longueur : Longueur minimale de 1. Longueur maximale de 1024.

Obligatoire : Non

TimeoutInMinutes

Durée qui peut s'écouler avant que l'état de la pile ne devienne `CREATE_FAILED` . Si n`DisableRollback`'est pas défini ou est défini sur `false` , la pile sera annulée.

Type: Integer

Plage valide : Valeur minimale est 1.

Obligatoire : Non

Outputs

StackId

Identifiant unique de la pile.

Type: String (chaîne)

StackStatus

Statut actuel de la pile.

Type: String (chaîne)

Valeurs valides: `CREATE_IN_PROGRESS | CREATE_FAILED | CREATE_COMPLETE | ROLLBACK_IN_PROGRESS | ROLLBACK_FAILED | ROLLBACK_COMPLETE | DELETE_IN_PROGRESS | DELETE_FAILED | DELETE_COMPLETE | UPDATE_IN_PROGRESS | UPDATE_COMPLETE_CLEANUP_IN_PROGRESS | UPDATE_COMPLETE | UPDATE_ROLLBACK_IN_PROGRESS | UPDATE_ROLLBACK_FAILED |`

`UPDATE_ROLLBACK_COMPLETE_CLEANUP_IN_PROGRESS | UPDATE_ROLLBACK_COMPLETE | REVIEW_IN_PROGRESS`

Obligatoire : Oui

StackStatusReason

Message d'échec ou de succès associé au statut de la pile.

Type: String (chaîne)

Obligatoire : Non

Pour plus d'informations, consultez CreateStack .

Considérations sur la sécurité

Avant de pouvoir utiliser l'action `aws:createStack`, vous devez attribuer la stratégie suivante au rôle responsable de l'automatisation d'IAM. Pour de plus amples informations sur le rôle de responsable, veuillez consulter Tâche 1 : Créer un rôle de service pour Automation (p. 310).

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "sqs:*",
            "cloudformation:CreateStack",
            "cloudformation:DescribeStacks"
         ],
         "Resource":"*"
      }
   ]
}
```

aws:createTags – Créer des balises pour les ressources AWS

Créez de nouvelles balises pour les instances EC2 ou les instances gérées Systems Manager.

Input

Cette action prend en charge la plupart des CreateTags paramètres EC2 AddTagsToResource et SSM. Pour de plus amples informations, veuillez consulter CreateTags et AddTagsToResource.

L'exemple suivant montre comment baliser une AMI et une instance comme étant des ressources de production pour un service particulier.

YAML

```
name: createTags
action: aws:createTags
maxAttempts: 3
onFailure: Abort
inputs:
  ResourceType: EC2
  ResourceIds:
  - ami-9a3768fa
  - i-02951acd5111a8169
  Tags:
```

```
  - Key: production
    Value: ''
  - Key: department
    Value: devops
```

JSON

```
{
    "name": "createTags",
    "action": "aws:createTags",
    "maxAttempts": 3,
    "onFailure": "Abort",
    "inputs": {
        "ResourceType": "EC2",
        "ResourceIds": [
            "ami-9a3768fa",
            "i-02951acd5111a8169"
        ],
        "Tags": [
            {
                "Key": "production",
                "Value": ""
            },
            {
                "Key": "department",
                "Value": "devops"
            }
        ]
    }
}
```

ResourceIds

des ressources à baliser.IDs Si le type de ressources n'est pas « EC2 », ce champ peut uniquement contenir un seul élément.

Type: Liste de chaînes

Obligatoire : Oui

Tags

Balises à associer aux ressources.

Type: Liste des mappages

Obligatoire : Oui

ResourceType

Type des ressources qui doivent être balisées. Si le type n'est pas indiqué, la valeur par défaut « EC2 » est utilisée.

Type: String (chaîne)

Obligatoire : Non

Valeurs valides: `EC2| | |ManagedInstanceMaintenanceWindowParameter`

Output

None (Aucun)

aws:deleteImage – Supprimer une Amazon Machine Image

Supprime l'image spécifiée et tous les instantanés associés.

Input

Cette action prend en charge un seul paramètre. Pour plus d'informations, consultez la documentation pour DeregisterImage et DeleteSnapshot .

YAML

```
name: deleteMyImage
action: aws:deleteImage
maxAttempts: 3
timeoutSeconds: 180
onFailure: Abort
inputs:
  ImageId: ami-12345678
```

JSON

```
{
    "name": "deleteMyImage",
    "action": "aws:deleteImage",
    "maxAttempts": 3,
    "timeoutSeconds": 180,
    "onFailure": "Abort",
    "inputs": {
        "ImageId": "ami-12345678"
    }
}
```

ImageId

ID de l'image à supprimer.

Type: String (chaîne)

Obligatoire : Oui

Output

None (Aucun)

aws:deleteStack – Supprimer une pile AWS CloudFormation

Supprime une pile AWS CloudFormation.

Entrées

YAML

```
name: deleteStack
action: aws:deleteStack
maxAttempts: 1
onFailure: Abort
inputs:
  StackName: "{{stackName}}"
```

JSON

```
{
   "name":"deleteStack",
   "action":"aws:deleteStack",
   "maxAttempts":1,
   "onFailure":"Abort",
   "inputs":{
      "StackName":"{{stackName}}"
   }
}
```

ClientRequestToken

Identifiant unique de cette règle `DeleteStack`. Spécifiez ce jeton si vous pensez réessayer les requêtes pour que AWS CloudFormation sache que vous ne tentez pas de supprimer une pile avec le même nom. Vous pouvez réessayer les requêtes `DeleteStack` pour vérifier que AWS CloudFormation les a reçues.

Type: String (chaîne)

Contraintes de longueur : Longueur minimale de 1. Longueur maximale de 128.

Modèle : [a-zA-Z][-a-zA-Z0-9]*

Obligatoire : Non

RetainResources.membre.N

Cette entrée s'applique uniquement aux piles qui ont un statut `DELETE_FAILED`. Une liste de ressources logiqueIDss pour les ressources que vous souhaitez conserver. Pendant la suppression, AWS CloudFormation supprime la pile, mais ne supprime pas les ressources conservées.

La conservation des ressources s'avère utile lorsque vous ne pouvez pas supprimer une ressource, telle qu'un compartiment S3 non vide, mais que vous voulez supprimer la pile.

Type : tableau de chaînes

Obligatoire : Non

RoleARN

Amazon Resource Name (ARN) d'un rôle IAM endossé par AWS CloudFormation pour créer la pile. AWS CloudFormation utilise les informations d'identification du rôle pour effectuer des appels en votre nom. AWS CloudFormation utilise toujours ce rôle pour toutes les opérations futures au niveau de la pile. Tant que les utilisateurs ont l'autorisation d'opérer sur la pile, AWS CloudFormation utilise ce rôle même si les utilisateurs n'ont pas l'autorisation de le transmettre. Vérifiez que le rôle accorde le plus faible nombre de privilèges.

Si vous ne spécifiez pas de valeur, AWS CloudFormation utilise le rôle qui était précédemment associé à la pile. Si aucun rôle n'est disponible, AWS CloudFormation utilise une session temporaire qui est générée à partir de vos informations d'identification utilisateur.

Type: String (chaîne)

Contraintes de longueur : Longueur minimale de 20. Longueur maximale de 2048.

Obligatoire : Non

StackName

Le nom ou l'ID de la pile unique auquel la pile est associée.

Type: String (chaîne)

Obligatoire : Oui

Considérations sur la sécurité

Avant de pouvoir utiliser l'action `aws:deleteStack`, vous devez attribuer la stratégie suivante au rôle responsable de l'automatisation d'IAM. Pour de plus amples informations sur le rôle de responsable, veuillez consulter Tâche 1 : Créer un rôle de service pour Automation (p. 310).

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "sqs:*",
            "cloudformation:DeleteStack",
            "cloudformation:DescribeStacks"
         ],
         "Resource":"*"
      }
   ]
}
```

aws:executeAutomation – Exécuter une autre exécution d'automatisation

Exécute un flux de travail Automation secondaire en appelant un document Automation secondaire. Avec cette action, vous pouvez créer des documents Automation pour la plupart de vos flux de travail courants et faire référence à ces documents pendant une exécution d'Automation. Cette action peut simplifier vos documents Automation en supprimant la nécessité de dupliquer les étapes sur les documents similaires.

L'automatisation secondaire s'exécute dans le cadre de l'utilisateur qui a lancé l'automatisation principale. Cela signifie que l'automatisation secondaire utilise les mêmes rôle IAM ou compte utilisateur que l'utilisateur qui a commencé la première automatisation.

> **Important**
>
> Si vous spécifiez les paramètres d'une automatisation secondaire qui utilise un rôle de responsable (un rôle ayant recours à la stratégie iam:passRole), l'utilisateur ou le rôle qui a lancé l'automatisation principale doit donc avoir l'autorisation de transférer ce rôle de responsable dans l'automatisation secondaire. Pour de plus amples informations sur la configuration d'un rôle de responsable pour l'automatisation, veuillez consulter Méthode 2 : Utiliser IAM pour configurer des rôles pour Automation (p. 310).

Entrées

YAML

```
name: Secondary_Automation_Workflow
action: aws:executeAutomation
maxAttempts: 3
timeoutSeconds: 3600
onFailure: Abort
inputs:
  DocumentName: secondaryWorkflow
  RuntimeParameters:
    instanceIds:
```

```
    - i-1234567890abcdef0
```

JSON

```
{
   "name":"Secondary_Automation_Workflow",
   "action":"aws:executeAutomation",
   "maxAttempts":3,
   "timeoutSeconds":3600,
   "onFailure":"Abort",
   "inputs":{
      "DocumentName":"secondaryWorkflow",
      "RuntimeParameters":{
         "instanceIds":[
            "i-1234567890abcdef0"
         ]
      }
   }
}
```

DocumentName

Le nom du document Automation secondaire à exécuter pendant cette étape. Le document doit appartenir au même compte AWS que le document Automation principal.

Type: String (chaîne)

Obligatoire : Oui

DocumentVersion

La version du document Automation secondaire à exécuter. Si elle n'est pas spécifiée, l'automatisation exécute la version du document par défaut.

Type: String (chaîne)

Obligatoire : Oui

RuntimeParameters

Paramètres obligatoires pour l'exécution du document secondaire. Le mappage utilise le format suivant : {"parameter1" : ["value1"], "parameter2" : ["value2"] }

Type: Map

Obligatoire : Non

Output

Output

La sortie générée par l'exécution secondaire. Vous pouvez référencer la sortie à l'aide du format suivant : *`Secondary_Automation_Step_Name`*Sortie

Type: StringList

Voici un exemple :

```
- name: launchNewWindowsInstance
  action: 'aws:executeAutomation'
  onFailure: Abort
```

```
  inputs:
    DocumentName: launchWindowsInstance
  nextStep: getNewInstanceRootVolume
- name: getNewInstanceRootVolume
  action: 'aws:executeAwsApi'
  onFailure: Abort
  inputs:
    Service: ec2
    Api: DescribeVolumes
    Filters:
    - Name: attachment.device
        Values:
        - /dev/sda1
    - Name: attachment.instance-id
        Values:
        - '{{launchNewWindowsInstance.Output}}'
  outputs:
  - Name: rootVolumeId
    Selector: '$.Volumes[0].VolumeId'
    Type: String
  nextStep: snapshotRootVolume
- name: snapshotRootVolume
  action: 'aws:executeAutomation'
  onFailure: Abort
  inputs:
    DocumentName: AWS-CreateSnapshot
    RuntimeParameters:
    VolumeId:
    - '{{getNewInstanceRootVolume.rootVolumeId}}'
    Description:
    - 'Initial root snapshot for {{launchNewWindowsInstance.Output}}'
```

ExecutionId

L'ID d'exécution de l'exécution secondaire.

Type: String (chaîne)

Statut

L'état de l'exécution secondaire.

Type: String (chaîne)

aws:executeAwsApi – Appeler et exécuter des actions de l'API AWS

Appels et exécute des actions d'API AWS. La plupart des actions d'API sont prises en charge, mais pas toutes les actions d'API ont été testées. Par exemple, les actions d'API suivantes sont prises en charge : CreateImage, supprimer un compartiment RebootDBInstance , et CreateGroups , pour n'en citer que quelques-uns. Les actions API de streaming telles que l'action Get Object ne sont pas prises en charge. Pour de plus amples informations et des exemples sur l'utilisation de cette action, veuillez consulter Appel d'autres services AWS à partir d'un flux de travail Systems Manager Automation (p. 470).

Input

Les entrées sont définies par l'action d'API que vous choisissez.

YAML

```
action: aws:executeAwsApi
```

```
inputs:
  Service: The official namespace of the service
  Api: The API action or method name
  API action inputs or parameters: A value
outputs: # These are user-specified outputs
- Name: The name for a user-specified output key
  Selector: A response object specified by using jsonpath format
  Type: The data type
```

JSON

```
{
   "action":"aws:executeAwsApi",
   "inputs":{
      "Service":"The official namespace of the service",
      "Api":"The API action or method name",
      "API action inputs or parameters":"A value"
   },
   "outputs":[ These are user-specified outputs
      {
         "Name":"The name for a user-specified output key",
         "Selector":"A response object specified by using JSONPath format",
         "Type":"The data type"
      }
   ]
}
```

Service

L'espace de noms de service AWS qui contient l'action d'API que vous souhaitez exécuter. Vous pouvez afficher la liste des espaces de noms des services AWS pris en charge dans la section Services disponibles du Kit SDK AWS pour Python (Boto3). L'espace de noms se trouve dans la section Client . Par exemple, l'espace de noms pour Systems Manager est `ssm` . L'espace de noms pour Amazon EC2 est `ec2` .

Type: String (chaîne)

Obligatoire : Oui

Api

Le nom de l'action d'API que vous voulez exécuter. Vous pouvez afficher les actions d'API (également appelées méthodes) en choisissant un service dans le panneau de navigation de gauche sur la page Services Reference (Référence des services) suivante. Choisissez une méthode dans la section Client pour le service que vous voulez appeler. Par exemple, toutes les actions d'API (méthodes) pour Amazon RDS sont répertoriées sur la page suivante : Méthodes Amazon RDS.

Type: String (chaîne)

Obligatoire : Oui

Entrées d'action d'API

Une ou plusieurs entrées d'action d'API. Vous pouvez afficher les entrées disponibles (également appelées paramètres) en choisissant un service dans le panneau de navigation de gauche sur la page Services Reference (Référence des services) suivante. Choisissez une méthode dans la section Client pour le service que vous voulez appeler. Par exemple, toutes les méthodes pour Amazon RDS sont répertoriées sur la page suivante : Méthodes Amazon RDS. Choisissez la méthode describe_db_instances et faites défiler vers le bas pour voir les paramètres disponibles, tels que DBInstanceIdentifier , Name et Values.

YAML

```
inputs:
  Service: The official namespace of the service
  Api: The API action name
  API input 1: A value
  API Input 2: A value
  API Input 3: A value
```

JSON

```
"inputs":{
      "Service":"The official namespace of the service",
      "Api":"The API action name",
      "API input 1":"A value",
      "API Input 2":"A value",
      "API Input 3":"A value"
}
```

Type: Déterminé par l'action d'API choisie

Obligatoire : Oui

Nom

Nom de la sortie.

Type: String (chaîne)

Obligatoire : Oui

Selector

L'attribut JSONPath spécifique dans l'objet de réponse. Vous pouvez afficher les objets de réponse en choisissant un service dans le panneau de navigation de gauche sur la page Services Reference (Référence des services) suivante. Choisissez une méthode dans la section Client pour le service que vous voulez appeler. Par exemple, toutes les méthodes pour Amazon RDS sont répertoriées sur la page suivante : Méthodes Amazon RDS. Choisissez la méthode describe_db_instances et faites défiler la page jusqu'à la section Response Structure (Structure d'une réponse). DBInstancesest répertorié en tant qu'objet de réponse.

Type: Entier, Booléen, Chaîne, StringList , StringMap , ouMapList

Obligatoire : Oui

Tapez

Type de données de l'élément de réponse.

Type: Varie

Obligatoire : Oui

aws:executeScript – Exécuter un script

Exécute le Python ou le PowerShell script fourni, à l'aide de l'environnement d'exécution et du gestionnaire spécifiés. (Pour PowerShell , le gestionnaire n'est pas requis.)

Actuellement, l'`aws:executeScript`action contient les modules PowerShell principaux préinstallés suivants.

- Microsoft.PowerShell.Host
- Microsoft.PowerShell.Management
- Microsoft.PowerShell.Security
- Microsoft.PowerShell.Utility
- PackageManagement
- PowerShellGet

Pour utiliser des modules PowerShell Core qui ne sont pas préinstallés, votre script doit installer le module avec l'`-Force`indicateur , comme illustré dans la commande suivante.

```
Install-Module ModuleName -Force
```

Pour utiliser les applets PowerShell de commande Core dans votre script, nous vous recommandons d'utiliser les `AWS.Tools` modules, comme illustré dans les commandes suivantes.

> **Important**
>
> L'installation du module `AWSPowerShell.NetCore` n'est pas prise en charge.

- Applets de commande Amazon S3.

```
Install-Module AWS.Tools.S3 -Force
Get-S3Bucket -BucketName bucketname
```

- Applets de commande Amazon EC2.

```
Install-Module AWS.Tools.EC2 -Force
Get-EC2InstanceStatus -InstanceId instanceId
```

- Applets de commande Outils AWS pour Windows PowerShell communs ou indépendants du service.

```
Install-Module AWS.Tools.Common -Force
Get-AWSRegion
```

Si votre script initialise de nouveaux objets en plus d'utiliser les applets PowerShell de commande Core, vous devez également importer le module comme illustré dans la commande suivante.

```
Install-Module AWS.Tools.EC2 -Force
Import-Module AWS.Tools.EC2

$tag = New-Object Amazon.EC2.Model.Tag
$tag.Key = "myTag"
$tag.Value = "myTagValue"

New-EC2Tag -Resource i-12345678 -Tag $tag
```

Pour obtenir des exemples d'installation et d'importation de `AWS.Tools` modules, et d'utilisation des applets de commande PowerShell Core dans le contenu d'un document Automation, consultez Procédure : Utilisation de Document Builder pour créer un document Automation personnalisé (p. 670) .

> **Note**
>
> Chaque `aws:executeScript` action peut s'exécuter jusqu'à une durée maximale de 600 secondes (ten minutes). Vous pouvez limiter le délai d'attente en spécifiant le `timeoutSeconds` paramètre d'une `aws:executeScript` étape.

Input

Fournissez l'environnement d'exécution et le gestionnaire requis pour exécuter le script Python 3.6, Python 3.7 ou PowerShell Core 6.0 fourni.

> **Important**
>
> Le paramètre d'entrée de script n'est pas pris en charge pour les documents JSON. Les documents JSON doivent fournir du contenu de script à l'aide du paramètre d'entrée de pièce jointe.

YAML

```
action: "aws:executeScript"
inputs:
 Runtime: "python3.6"
 Handler: "script_handler"
 InputPayload:
  "parameter1": "parameter_value1"
  "parameter2": "parameter_value2"
 Script:
  -
   "def script_handler(events, context):"
  -
   "(script commands)"
 Attachment: "zip-file-name-1.zip"
```

JSON

```
{
    "action": "aws:executeScript",
    "inputs": {
        "Runtime": "python3.6",
        "Handler": "script_handler",
        "InputPayload": {
            "parameter1": "parameter_value1",
            "parameter2": "parameter_value2"
        },
        "Attachment": "zip-file-name-1.zip"
    }
}
```

Exécution

Langue d'exécution à utiliser pour exécuter le script fourni. Actuellement, aws:executeScript prend en charge Python 3.6 (python3.6), Python 3.7 (python3.7) et PowerShell Core 6.0 (dotnetcore2.1)

Valeurs prises en charge : **`python3.6`**| | |**`python3.7PowerShell Core 6.0`**

Type: String (chaîne)

Obligatoire : Oui

Handler

Entrée pour l'exécution du script, généralement un nom de fonction. Vous devez vous assurer que la fonction définie dans le gestionnaire possède deux paramètres, `events` et `context` . (Non requis pour PowerShell .)

Type: String (chaîne)

Obligatoire : Oui (Python) | Non (PowerShell)

InputPayload

Objet JSON ou YAML qui sera transmis au premier paramètre du gestionnaire. Il peut être utilisé pour transmettre des données d'entrée au script.

Type: String (chaîne)

Obligatoire : Non

Script

Script incorporé que vous souhaitez exécuter pendant l'exécution de l'automatisation. (Non pris en charge pour les documents JSON.)

Type: String (chaîne)

Obligatoire : Non (Python) | Oui (PowerShell)

Attachment

Nom d'un fichier de script autonome ou .zip pouvant être appelé par l'action. Pour appeler un fichier pour Python, utilisez le `filename.method_name` format dans `Handler` . Pour PowerShell , appelez la pièce jointe à l'aide d'un script en ligne. Gzip n'est pas pris en charge.

Lorsque vous incluez des bibliothèques Python dans votre attachement, nous vous recommandons d'ajouter un `__init__.py` fichier vide dans chaque répertoire du module. Cela vous permet d'importer les modules de la bibliothèque dans votre pièce jointe au sein du contenu de votre script. Par exemple: `from library import module`

Type: String (chaîne)

Obligatoire : Non

Output

Charge utile

Représentation JSON de l'objet renvoyé par votre fonction .

aws:executeStateMachine – Exécuter une machine d'état AWS Step Functions

Exécutez une machine d'état AWS Step Functions.

Entrées

Cette action prend en charge la plupart des paramètres pour l'action Step Functions StartExecution d'API.

YAML

```
name: executeTheStateMachine
action: aws:executeStateMachine
inputs:
  stateMachineArn: StateMachine_ARN
  input: '{"parameters":"values"}'
  name: name
```

JSON

```
{
    "name": "executeTheStateMachine",
    "action": "aws:executeStateMachine",
    "inputs": {
        "stateMachineArn": "StateMachine_ARN",
        "input": "{\"parameters\":\"values\"}",
        "name": "name"
    }
}
```

stateMachineArn

ARN de la machine d'état Step Functions.

Type: String (chaîne)

Obligatoire : Oui

nom

Nom de l'exécution.

Type: String (chaîne)

Obligatoire : Non

input

Chaîne qui contient les données d'entrée JSON pour l'exécution.

Type: String (chaîne)

Obligatoire : Non

aws:invokeLambdaFunction – Appeler une fonction AWS Lambda

Appelle la fonction Lambda spécifiée.

> **Note**
>
> Chaque `aws:invokeLambdaFunction` action peut s'exécuter jusqu'à une durée maximale de 300 secondes (cinq minutes). Vous pouvez limiter le délai d'attente en spécifiant le `timeoutSeconds` paramètre d'une `aws:invokeLambdaFunction` étape.

Input

Cette action prend en charge la plupart des paramètres invoqués du service Lambda. Pour plus d'informations, consultez Invoquer.

YAML

```
name: invokeMyLambdaFunction
action: aws:invokeLambdaFunction
maxAttempts: 3
timeoutSeconds: 120
onFailure: Abort
inputs:
```

```
  FunctionName: MyLambdaFunction
```

JSON

```
{
    "name": "invokeMyLambdaFunction",
    "action": "aws:invokeLambdaFunction",
    "maxAttempts": 3,
    "timeoutSeconds": 120,
    "onFailure": "Abort",
    "inputs": {
        "FunctionName": "MyLambdaFunction"
    }
}
```

FunctionName

Le nom de la fonction Lambda. Cette fonction doit exister.

Type: String (chaîne)

Obligatoire : Oui

Qualificateur

Nom de version ou d'alias de la fonction.

Type: String (chaîne)

Obligatoire : Non

InvocationType

Type d'appel. La valeur par défaut est `RequestResponse`.

Type: String (chaîne)

Valeurs valides : `Event`| | |`RequestResponseDryRun`

Obligatoire : Non

LogType

Si `Tail`, le type d'appel doit être `RequestResponse`. AWS Lambda retourne les 4 derniers Ko de données de journal générés par votre fonction Lambda, codés en base64.

Type: String (chaîne)

Valeurs valides : `None`|`Tail`

Obligatoire : Non

ClientContext

Informations spécifiques au client.

Obligatoire : Non

Charge utile

Entrée JSON pour votre fonction Lambda.

Obligatoire : Non

Output

StatusCode

Code de statut d'exécution de la fonction.

FunctionError

Indique si une erreur s'est produite pendant l'exécution de la fonction Lambda. Si une erreur s'est produite, ce champ affiche les `Handled` erreurs `Unhandled` ou `Handled` . Les erreurs sont signalées par la fonction . Les `Unhandled` erreurs sont détectées et signalées par AWS Lambda .

LogResult

Journaux codés en base 64 pour l'appel de fonction Lambda. Des journaux sont présents uniquement si le type d'appel est `RequestResponse` et que des journaux ont été demandés.

Charge utile

Représentation JSON de l'objet renvoyé par la fonction Lambda. La charge utile est présente uniquement si le type d'appel est `RequestResponse`.

aws:pause – Suspendre une exécution d'automatisation

Cette action suspend l'exécution d'Automation. Une fois l'exécution interrompue, son statut est En attente. Pour poursuivre l'exécution d'Automation, utilisez l'action SendAutomationSignal d'API avec le type de signal Reprendre.

Input

L'entrée est comme suit.

YAML

```
name: pauseThis
action: aws:pause
inputs: {}
```

JSON

```
{
    "name": "pauseThis",
    "action": "aws:pause",
    "inputs": {}
}
```

Output

None (Aucun)

aws:runCommand – Exécuter une commande sur une instance gérée

Exécute les commandes spécifiées.

Note

L'automatisation prend uniquement en charge la sortie d'une action Fonctionnalité Exécuter la commande. Un document peut inclure plusieurs actions Fonctionnalité Exécuter la commande, mais la sortie est prise en charge pour une seule action à la fois.

Input

Cette action prend en charge la plupart des paramètres de la commande d'envoi. Pour plus d'informations, consultez SendCommand .

YAML

```
- name: checkMembership
  action: 'aws:runCommand'
  inputs:
    DocumentName: AWS-RunPowerShellScript
    InstanceIds:
      - '{{InstanceIds}}'
    Parameters:
      commands:
        - (Get-WmiObject -Class Win32_ComputerSystem).PartOfDomain
```

JSON

```
{
    "name": "checkMembership",
    "action": "aws:runCommand",
    "inputs": {
        "DocumentName": "AWS-RunPowerShellScript",
        "InstanceIds": [
            "{{InstanceIds}}"
        ],
        "Parameters": {
            "commands": [
                "(Get-WmiObject -Class Win32_ComputerSystem).PartOfDomain"
            ]
        }
    }
}
```

DocumentName

Nom du document Fonctionnalité Exécuter la commande.

Type: String (chaîne)

Obligatoire : Oui

InstanceIds

Instance IDs dans laquelle vous voulez que la commande s'exécute. Vous pouvez spécifier un maximum de 50 IDs .

Vous pouvez également utiliser le pseudo-paramètre `{{RESOURCE_ID}}` à la place de l'instance pour exécuter plus facilement la commande sur toutes les instances du groupe cible.IDs Pour de plus amples informations sur les pseudo-paramètres, veuillez consulter À propos des pseudo-paramètres (p. 736).

Une autre alternative consiste à envoyer des commandes à un parc d'instances à l'aide du paramètre Targets. Le paramètre Targets accepte les balises Amazon EC2. Pour de plus amples informations sur

l'utilisation du paramètre Targets, veuillez consulter Utilisation des contrôles de cibles et de taux pour envoyer des commandes à un parc (p. 955).

Type: StringList

Obligatoire : Non (si vous ne spécifiez pas InstanceIds ou n'utilisez pas le `{{RESOURCE_ID}}` pseudo-paramètre, vous devez spécifier le paramètre Targets.)

Cibles

Ensemble de critères de recherche qui cible les instances en utilisant une combinaison clé/valeur que vous spécifiez. Les cibles sont obligatoires si vous ne fournissez pas une ou plusieurs instances IDs dans l'appel. Pour de plus amples informations sur l'utilisation du paramètre Targets, veuillez consulter Utilisation des contrôles de cibles et de taux pour envoyer des commandes à un parc (p. 955).

Type: MapList(Le schéma du mappage dans la liste doit correspondre à l'objet. Pour de plus amples informations, veuillez consulter Target dans le document Référence d'API AWS Systems Manager.

Obligatoire : Non (si vous ne spécifiez pas de cibles, vous devez spécifier InstanceIds ou utiliser le `{{RESOURCE_ID}}` pseudo-paramètre.)

Voici un exemple :

YAML

```
- name: checkMembership
  action: aws:runCommand
  inputs:
    DocumentName: AWS-RunPowerShellScript
    Targets:
      - Key: tag:Stage
        Values:
          - Gamma
          - Beta
      - Key: tag-key
        Values:
          - Suite
    Parameters:
      commands:
        - (Get-WmiObject -Class Win32_ComputerSystem).PartOfDomain
```

JSON

```
{
    "name": "checkMembership",
    "action": "aws:runCommand",
    "inputs": {
        "DocumentName": "AWS-RunPowerShellScript",
        "Targets": [
            {
                "Key": "tag:Stage",
                "Values": [
                    "Gamma", "Beta"
                ]
            },
            {
                "Key": "tag:Application",
                "Values": [
                    "Suite"
                ]
            }
        ],
        "Parameters": {
```

```
            "commands": [
                "(Get-WmiObject -Class Win32_ComputerSystem).PartOfDomain"
            ]
        }
    }
}
```

Parameters

Paramètres obligatoires et facultatifs spécifiés dans le document.

Type: Map

Obligatoire : Non

CloudWatchOutputConfig

Options de configuration pour l'envoi de la sortie générée par la commande à Amazon CloudWatch Logs. Pour de plus amples informations sur l'envoi de la sortie d'une commande à CloudWatch Logs, veuillez consulter Configuration d'Amazon CloudWatch Logs pour Fonctionnalité Exécuter la commande (p. 1338).

Type: StringMap(Le schéma de la carte doit correspondre à l'objet. Pour plus d'informations, consultez CloudWatchOutputConfig dans le Référence d'API AWS Systems Manager .

Obligatoire : Non

Voici un exemple :

YAML

```
- name: checkMembership
  action: aws:runCommand
  inputs:
    DocumentName: AWS-RunPowerShellScript
    InstanceIds:
      - "{{InstanceIds}}"
    Parameters:
      commands:
        - "(Get-WmiObject -Class Win32_ComputerSystem).PartOfDomain"
    CloudWatchOutputConfig:
      CloudWatchLogGroupName: CloudWatchGroupForSSMAutomationService
      CloudWatchOutputEnabled: true
```

JSON

```
{
    "name": "checkMembership",
    "action": "aws:runCommand",
    "inputs": {
        "DocumentName": "AWS-RunPowerShellScript",
        "InstanceIds": [
            "{{InstanceIds}}"
        ],
        "Parameters": {
            "commands": [
                "(Get-WmiObject -Class Win32_ComputerSystem).PartOfDomain"
            ]
        },
        "CloudWatchOutputConfig" : {
                "CloudWatchLogGroupName": "CloudWatchGroupForSSMAutomationService",
                "CloudWatchOutputEnabled": true
        }
```

```
    }
}
```

Commentaire

Informations définies par l'utilisateur au sujet de la commande.

Type: String (chaîne)

Obligatoire : Non

DocumentHash

Hachage pour le document.

Type: String (chaîne)

Obligatoire : Non

DocumentHashType

Type de hachage.

Type: String (chaîne)

Valeurs valides : `Sha256|Sha1`

Obligatoire : Non

NotificationConfig

Configurations d'envoi des notifications.

Obligatoire : Non

OutputS3BucketName

Nom du compartiment S3 pour les réponses d'exécution de la commande.

Type: String (chaîne)

Obligatoire : Non

OutputS3KeyPrefix

Préfixe.

Type: String (chaîne)

Obligatoire : Non

ServiceRoleArn

ARN du rôle IAM.

Type: String (chaîne)

Obligatoire : Non

TimeoutSeconds

Durée d'attente en secondes pour qu'une commande soit livrée à l' Agent SSM sur une instance. Si la commande n'est pas reçue par l' Agent SSM sur l'instance avant que la valeur spécifiée ne soit atteinte, le statut de la commande passe à `Delivery Timed Out` .

Type: Integer

Obligatoire : Non

Output

CommandId

ID de la commande.

Statut

Statut de la commande.

ResponseCode

Code de réponse de la commande.

Output

Sortie de la commande.

aws:runInstances – Lancer une instance EC2

Lance une nouvelle instance.

Input

L'action prend en charge la plupart des paramètres de l'API. Pour plus d'informations, consultez la documentation de lRunInstances'API.

YAML

```
name: launchInstance
action: aws:runInstances
maxAttempts: 3
timeoutSeconds: 1200
onFailure: Abort
inputs:
  ImageId: ami-12345678
  InstanceType: t2.micro
  MinInstanceCount: 1
  MaxInstanceCount: 1
  IamInstanceProfileName: myRunCmdRole
  TagSpecifications:
  - ResourceType: instance
    Tags:
    - Key: LaunchedBy
      Value: SSMAutomation
    - Key: Category
      Value: HighAvailabilityFleetHost
```

JSON

```
{
   "name":"launchInstance",
   "action":"aws:runInstances",
   "maxAttempts":3,
   "timeoutSeconds":1200,
   "onFailure":"Abort",
   "inputs":{
```

```
      "ImageId":"ami-12345678",
      "InstanceType":"t2.micro",
      "MinInstanceCount":1,
      "MaxInstanceCount":1,
      "IamInstanceProfileName":"myRunCmdRole",
      "TagSpecifications":[
         {
            "ResourceType":"instance",
            "Tags":[
               {
                  "Key":"LaunchedBy",
                  "Value":"SSMAutomation"
               },
               {
                  "Key":"Category",
                  "Value":"HighAvailabilityFleetHost"
               }
            ]
         }
      ]
   }
}
```

ImageId

ID de l'Amazon Machine Image (AMI).

Type: String (chaîne)

Obligatoire : Oui

InstanceType

Type d'instance .

> **Note**
>
> Si aucune valeur de type d'instance n'est fournie, le type d'instance m1.small est utilisé.

Type: String (chaîne)

Obligatoire : Non

MinInstanceCount

Nombre minimum d'instances à lancer.

Type: String (chaîne)

Obligatoire : Non

MaxInstanceCount

Nombre maximum d'instances à lancer.

Type: String (chaîne)

Obligatoire : Non

AdditionalInfo

Instances réservées.

Type: String (chaîne)

Obligatoire : Non

BlockDeviceMappings

Périphériques de stockage en mode bloc pour l'instance.

Type: MapList

Obligatoire : Non

ClientToken

Identifiant permettant de garantir l'idempotence de la demande.

Type: String (chaîne)

Obligatoire : Non

DisableApiTermination

Active ou désactive la fin de l'API de l'instance.

Type: Booléen

Obligatoire : Non

EbsOptimized

Active ou désactive l'optimisation EBS.

Type: Booléen

Obligatoire : Non

IamInstanceProfileArn

ARN du profil d'instance IAM pour l'instance.

Type: String (chaîne)

Obligatoire : Non

IamInstanceProfileName

Nom du profil d'instance IAM pour l'instance.

Type: String (chaîne)

Obligatoire : Non

InstanceInitiatedShutdownBehavior

Indique si l'instance s'arrête ou est résiliée lors de l'arrêt du système.

Type: String (chaîne)

Obligatoire : Non

KernelId

ID du noyau.

Type: String (chaîne)

Obligatoire : Non

KeyName

Nom de la paire de clés.

Type: String (chaîne)

Obligatoire : Non

MaxInstanceCount

Nombre maximum d'instances à filtrer lors de la recherche d'offres.

Type: Integer

Obligatoire : Non

MinInstanceCount

Nombre minimum d'instances à filtrer lors de la recherche d'offres.

Type: Integer

Obligatoire : Non

Surveillance

Active ou désactive la surveillance détaillée.

Type: Booléen

Obligatoire : Non

NetworkInterfaces

Interfaces réseau.

Type: MapList

Obligatoire : Non

Placement

Placement de l'instance.

Type: StringMap

Obligatoire : Non

PrivateIpAddress

Adresse principale.IPv4

Type: String (chaîne)

Obligatoire : Non

RamdiskId

ID du disque RAM.

Type: String (chaîne)

Obligatoire : Non

SecurityGroupIds

des groupes de sécurité de l'instance.IDs

Type: StringList

Obligatoire : Non

SecurityGroups

Noms des groupes de sécurité de l'instance.

Type: StringList

Obligatoire : Non

SubnetId

ID de sous-réseau.

Type: String (chaîne)

Obligatoire : Non

TagSpecifications

Balises à appliquer aux ressources lors du lancement. Vous pouvez uniquement baliser des instances et des volumes au moment du lancement. Les balises spécifiées sont appliquées à toutes les instances ou volumes qui sont créés lors du lancement. Pour ajouter des balises à une instance après qu'elle a été lancée, utilisez l'action aws:createTags – Créer des balises pour les ressources AWS (p. 410).

Type: MapList(Pour plus d'informations, consultez TagSpecification .)

Obligatoire : Non

UserData

Script d'exécution fourni en tant que valeur de littéral de chaîne. Si une valeur littérale est entrée, elle doit être codée en base64.

Type: String (chaîne)

Obligatoire : Non

Output

InstanceIds

de ces instances.IDs

InstanceStates

État actuel de l'instance.

aws:sleep – Retarder une exécution d'automatisation

Retarde l'exécution de l'automatisation pour une durée spécifiée. Cette action utilise le format date et heure de l'Organisation internationale de normalisation (ISO) 8601. Pour de plus amples informations sur ce format date et heure, veuillez consulter ISO 8601.

Input

Vous pouvez retarder l'exécution pour une durée spécifiée.

YAML

```
name: sleep
action: aws:sleep
inputs:
  Duration: PT10M
```

JSON

```
{
   "name":"sleep",
   "action":"aws:sleep",
   "inputs":{
      "Duration":"PT10M"
   }
}
```

Vous pouvez aussi retarder l'exécution jusqu'à une date et une heure spécifiées. Si les date et heure spécifiées sont passées, l'action est réalisée immédiatement.

YAML

```
name: sleep
action: aws:sleep
inputs:
  Timestamp: '2020-01-01T01:00:00Z'
```

JSON

```
{
    "name": "sleep",
    "action": "aws:sleep",
    "inputs": {
        "Timestamp": "2020-01-01T01:00:00Z"
    }
}
```

> **Note**
>
> Automation prend actuellement en charge un délai maximum de 604 800 secondes (7 jours).

Duration (Durée)

Une durée ISO 8601. Vous ne pouvez pas spécifier une durée négative.

Type: String (chaîne)

Obligatoire : Non

Horodatage

Un horodatage ISO 8601. Si vous ne spécifiez aucune valeur pour ce paramètre, vous devez alors spécifier une valeur pour le paramètre `Duration`.

Type: String (chaîne)

Obligatoire : Non

Output

None (Aucun)

aws:waitForAwsResourceProperty – Attendre une propriété de ressource AWS

L'action aws:waitForAwsResourceProperty permet à votre flux de travail d'automatisation d'attendre qu'une ressource ou un événement atteigne un état spécifique avant de poursuivre le flux de travail. Pour de plus amples informations et des exemples sur l'utilisation de cette action, veuillez consulter Appel d'autres services AWS à partir d'un flux de travail Systems Manager Automation (p. 470).

Input

Les entrées sont définies par l'action d'API que vous choisissez.

YAML

```
action: aws:waitForAwsResourceProperty
inputs:
  Service: The official namespace of the service
  Api: The API action or method name
  API action inputs or parameters: A value
  PropertySelector: Response object
  DesiredValues:
  - Desired property value
```

JSON

```
{
  "action": "aws:waitForAwsResourceProperty",
  "inputs": {
    "Service":"The official namespace of the service",
    "Api":"The API action or method name",
    "API action inputs or parameters":"A value",
    "PropertySelector": "Response object",
    "DesiredValues": [
      "Desired property value"
    ]
  }
}
```

Service

L'espace de noms de service AWS qui contient l'action d'API que vous souhaitez exécuter. Par exemple, l'espace de noms pour Systems Manager est `ssm` . L'espace de noms pour Amazon EC2 est `ec2` . Vous pouvez afficher une liste des espaces de noms de service AWS pris en charge dans la section Services disponibles du AWS CLI Command Reference .

Type: String (chaîne)

Obligatoire : Oui

Api

Le nom de l'action d'API que vous voulez exécuter. Vous pouvez afficher les actions d'API (également appelées méthodes) en choisissant un service dans le panneau de navigation de gauche sur la page Services Reference (Référence des services) suivante. Choisissez une méthode dans la section

Client pour le service que vous voulez appeler. Par exemple, toutes les actions d'API (méthodes) pour Amazon RDS sont répertoriées sur la page suivante : Méthodes Amazon RDS.

Type: String (chaîne)

Obligatoire : Oui

Entrées d'action d'API

Une ou plusieurs entrées d'action d'API. Vous pouvez afficher les entrées disponibles (également appelées paramètres) en choisissant un service dans le panneau de navigation de gauche sur la page Services Reference (Référence des services) suivante. Choisissez une méthode dans la section Client pour le service que vous voulez appeler. Par exemple, toutes les méthodes pour Amazon RDS sont répertoriées sur la page suivante : Méthodes Amazon RDS. Choisissez la méthode describe_db_instances et faites défiler vers le bas pour voir les paramètres disponibles, tels que DBInstanceIdentifier , Name et Values.

YAML

```
inputs:
  Service: The official namespace of the service
  Api: The API action name
  API input 1: A value
  API Input 2: A value
  API Input 3: A value
```

JSON

```
"inputs":{
      "Service":"The official namespace of the service",
      "Api":"The API action name",
      "API input 1":"A value",
      "API Input 2":"A value",
      "API Input 3":"A value"
}
```

Type: Déterminé par l'action d'API choisie

Obligatoire : Oui

PropertySelector

L'attribut JSONPath spécifique dans l'objet de réponse. Vous pouvez afficher les objets de réponse en choisissant un service dans le panneau de navigation de gauche sur la page Services Reference (Référence des services) suivante. Choisissez une méthode dans la section Client pour le service que vous voulez appeler. Par exemple, toutes les méthodes pour Amazon RDS sont répertoriées sur la page suivante : Méthodes Amazon RDS. Choisissez la méthode describe_db_instances et faites défiler la page jusqu'à la section Response Structure (Structure d'une réponse). DBInstancesest répertorié en tant qu'objet de réponse.

Type: Entier, Booléen, Chaîne, StringList , StringMap , ouMapList

Obligatoire : Oui

DesiredValues

État attendu ou état à partir duquel le flux de travail d'automatisation se poursuit.

Type: Varie

Obligatoire : Oui

Variables système Automation

Les documents Automation Systems Manager utilisent les variables suivantes. Pour obtenir un exemple d'utilisation de ces variables, affichez la source JSON du document `AWS-UpdateWindowsAmi`.

Pour afficher la source JSON du document AWS-UpdateWindowsAmi

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.
3. Dans la liste de documents, utilisez la barre de recherche ou les nombres à droite de la barre de recherche pour choisir le document AWS-UpdateWindowsAmi.
4. Choisissez l'onglet Content (Contenu).

Variables système

Les documents Automation prennent en charge actuellement les variables système suivantes.

Variable	Details (Détails)
`global:ACCOUNT_ID`	ID du compte AWS de l'utilisateur ou du rôle AWS Identity and Access Management (IAM) dans lequel Automation s'exécute.
`global:DATE`	Date (au moment de l'exécution) au format AAAA-MM-JJ.
`global:DATE_TIME`	Date et heure (au moment de l'exécution) au format AAAA-MM-dd_HH.mm.ss.
`global:REGION`	La région dans laquelle le document est exécuté. Par exemple, us-east-2.

Variables d'automatisation

Les documents Automation prennent en charge actuellement les variables d'automatisation.

Variable	Details (Détails)
`automation:EXECUTION_ID`	Identifiant unique attribué à l'exécution de l'automatisation effective. Par exemple, `1a2b3c-1a2b3c-1a2b3c-1a2b3c1a2b3c1a2b3c`.

Rubriques

- Terminology (p. 437)
- Scénarios pris en charge (p. 439)
- Scénarios non pris en charge (p. 443)

Terminology

Les conditions suivantes décrivent comment les variables et les paramètres sont résolus.

Durée	Définition	Exemple
ARN constant	ARN valide sans variables	`arn:aws:iam::123456789012:role/roleName`
Paramètre du document	Paramètre défini au niveau du document pour un document Automation (par exemple, `instanceId`). Le paramètre est utilisé dans un remplacement de chaîne de base. Sa valeur est fournie au moment du démarrage de l'exécution.	`{ "description": "Create Image Demo", "version": "0.3", "assumeRole": "Your_Automation_Assume_Role_ARN", "parameters":{ "instanceId": { "type": "String", "description": "Instance to create image from" } }`
Variable système	Variable générale remplacée dans le document lors de l'évaluation d'une partie du document.	`"activities": [{ "id": "copyImage", "activityType": "AWS-CopyImage", "maxAttempts": 1, "onFailure": "Continue", "inputs": { "ImageName": "{{imageName}}", "SourceImageId": "{{sourceImageId}}", "SourceRegion": "{{sourceRegion}}", "Encrypted": true, "ImageDescription": "Test CopyImage Description created on {{global:DATE}}" } }]`
Variable d'automatisation	Variable relative à l'exécution d'Automation remplacée dans le document lors de l'évaluation d'une partie du document.	`{ "name": "runFixedCmds", "action": "aws:runCommand", "maxAttempts": 1, "onFailure": "Continue", "inputs": { "DocumentName": "AWS-RunPowerShellScript", "InstanceIds": [ "{{LaunchInstance.InstanceIds}}" ], "Parameters": { "commands": [ "dir", "date",`

Durée	Définition	Exemple
		```""{{outputFormat}}" -f "left","right","{{global:DATE}}","{{auto] } } } } ```
Paramètre SSM	Une variable définie dans Parameter Store. Il ne peut pas être référencé directement dans l'entrée de l'étape. Des autorisations peuvent être requises pour accéder au paramètre.	```description: Launch new Windows test instance schemaVersion: '0.3' assumeRole: '{{AutomationAssumeRole}}' parameters: AutomationAssumeRole: type: String default: '' description: >- (Required) The ARN of the role that allows Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses your IAM permissions to run this document. LatestAmi: type: String default: >- {{ssm:/aws/service/ami-windows-latest/Windows_Server-2016-English-Full-Base}} description: The latest Windows Server 2016 AMI queried from the public parameter. mainSteps: - name: launchInstance action: 'aws:runInstances' maxAttempts: 3 timeoutSeconds: 1200 onFailure: Abort inputs: ImageId: '{{LatestAmi}}' ...```

Scénarios pris en charge

Scénario	Commentaires	Exemple
ARN `assumeRole` constant lors de la création.	Un contrôle d'autorisation est effectué pour vérifier que	```{ ```

Scénario	Commentaires	Exemple
	l'utilisateur appelant est autorisé à passer le `assumeRole` donné.	`"description": "Test all Automation resolvable parameters", "schemaVersion": "0.3", "assumeRole": "arn:aws:iam::123456789012 roleName", "parameters": { ...`
Paramètre de document fourni pour le`assumeRole` à l'exécution.	Doit être défini dans la liste des paramètres du document.	`{ "description": "Test all Automation resolvable parameters", "schemaVersion": "0.3", "assumeRole": "{{dynamicARN}}", "parameters": { ...`
Valeur fournie pour le paramètre du document au démarrage.	Le client fournit la valeur à utiliser pour un paramètre. Toutes les entrées d'exécution fournies au moment du démarrage doivent être définies dans la liste des paramètres du document.	`... "parameters": { "amiId": { "type": "String", "default": "ami-12345678", "description": "list of commands to run as part of first step" }, ...` Les entrées pour démarrer l'exécution d'Automation incluent : `{"amiId" : ["ami-12345678"] }`

Scénario	Commentaires	Exemple

Paramètre SSM référencé dans le contenu du document.	La variable existe dans le compte du client, ou est un paramètre accessible publiquement, et le `assumeRole` pour le document a accès à la variable. Un contrôle est effectué à la création pour confirmer que le `assumeRole` y a accès. Le paramètre ne peut pas être référencé directement dans l'entrée de l'étape.	`...`
		`parameters:`
		`  LatestAmi:`
		`  type: String`
		`  default: >-`
		`  {{ssm:/aws/service/ami-windows-latest/Windows_Server-2016-English-Full-Base}}`
		`  description: The latest Windows Server 2016 AMI queried from the public parameter.`
		`mainSteps:`
		`  - name: launchInstance`
		`  action: 'aws:runInstances'`
		`  maxAttempts: 3`
		`  timeoutSeconds: 1200`
		`  onFailure: Abort`
		`  inputs:`
		`  ImageId: '{{LatestAmi}}'`
		`...`

Variable système référencée dans la définition de l'étape	Une variable système est remplacée dans le document au moment de l'exécution. La valeur injectée dans le document est relative au moment où la substitution se produit. En d'autres termes, la valeur d'une variable de temps injectée à l'étape 1 est différente de la valeur injectée à l'étape 3 en raison du temps nécessaire pour exécuter les étapes entre elles. Les variables système n'ont pas besoin d'être définies dans la liste des paramètres du document.	`...`
		`  "mainSteps": [`
		`  {`
		`  "name": "RunSomeCommands",`
		`  "action": "aws:runCommand",`
		`  "maxAttempts": 1,`
		`  "onFailure": "Continue",`
		`  "inputs": {`
		`  "DocumentName": "AWS:RunPowerShell",`
		`  "InstanceIds": ["{{LaunchInstance.InstanceIds}}"],`
		`  "Parameters": {`
		`  "commands" : [`
		`  "echo {The time is now {{global:DATE_TIME}}}"`
		`  ]`
		`  }`
		`  }`
		`}, ...`

Scénario	Commentaires	Exemple

Variable d'automatisation référencée dans la définition de l'étape.	Les variables d'automatisation n'ont pas besoin d'être définies dans la liste des paramètres du document. La seule variable d'automatisation prise en charge est automation:EXECUTION_ID.	`...`
		`"mainSteps": [`
		`{`
		`"name":`
		`"invokeLambdaFunction",`
		`"action":`
		`"aws:invokeLambdaFunction",`
		`"maxAttempts": 1,`
		`"onFailure":`
		`"Continue",`
		`"inputs": {`
		`"FunctionName":`
		`"Hello-World-`
		`LambdaFunction",`
		
		`"Payload" :`
		`"{ "executionId" :`
		`"{{automation:EXECUTION_ID}}" }"`
		`}`
		`}`
		`...`

Reportez-vous à la sortie de l'étape précédente dans la définition de l'étape suivante.	Il s'agit de la redirection du paramètre. La sortie d'une étape précédente est référencée à l'aide de la syntaxe `{{stepName.OutputName}}`. Cette syntaxe ne peut pas être utilisée par le client pour les paramètres du document. Ceci est résolu au moment de l'exécution de l'étape de référence. Le paramètre n'est pas répertorié dans les paramètres du document.	`...`
		`"mainSteps": [`
		`{`
		`"name":`
		`"LaunchInstance",`
		`"action":`
		`"aws:runInstances",`
		`"maxAttempts": 1,`
		`"onFailure":`
		`"Continue",`
		`"inputs": {`
		`"ImageId":`
		`"{{amiId}}",`
		`"MinInstanceCount":`
		`1,`
		`"MaxInstanceCount":`
		`2`
		`}`
		`},`
		`{`
		`"name":"changeState",`
		`"action":`
		`"aws:changeInstanceState",`
		`"maxAttempts": 1,`
		`"onFailure":`
		`"Continue",`
		`"inputs": {`
		`"InstanceIds":`
		`["{{LaunchInstance.InstanceIds}}"],`
		`"DesiredState":`
		`"terminated"`
		`}`
		`}`
		
		`...`

Scénarios non pris en charge

Scénario	Commentaire	Exemple

Paramètre SSM fourni pour le `assumeRole` à la création	Non pris en charge.	`...`
		
		`{`
		`"description": "Test all Automation resolvable parameters",`
		`"schemaVersion": "0.3",`
		`"assumeRole": "{{ssm:administratorRoleARN}}",`
		`"parameters": {`
		
		`...`

Paramètre SSM directement référencé dans l'entrée de l'étape.	Renvoie une exception `InvalidDocumentContent` au moment de la création.	`...`
		`mainSteps:`
		`- name: launchInstance`
		`action: 'aws:runInstances'`
		`maxAttempts: 3`
		`timeoutSeconds: 1200`
		`onFailure: Abort`
		`inputs:`
		`ImageId: '{{ssm:/aws/service/ami-windows-latest/Windows_Server-2016-English-Full-Base}}'`
		`...`

Définition de l'étape des variables	La définition d'une étape du document est construite en variables.	`...`
		
		`"mainSteps": [`
		`{`
		`"name": "LaunchInstance",`
		`"action": "aws:runInstances",`
		`"{{attemptModel}}": 1,`
		`"onFailure": "Continue",`
		`"inputs": {`
		`"ImageId": "ami-12345678",`
		`"MinInstanceCount": 1,`
		`"MaxInstanceCount": 2`
		`}`
		
		`...`
		
		`User supplies input :`
		`{ "attemptModel" : "minAttempts" }`

Scénario	Commentaire	Exemple

Paramètres du document de références croisées	L'utilisateur fournit un paramètre d'entrée au moment du démarrage qui est une référence à un autre paramètre du document.	```
		...
		"parameters": {
		"amiId": {
		"type": "String",
		"default":
		"ami-7f2e6015",
		"description": "list
		of commands to run as part
		of first step"
		},
		"alternateAmiId": {
		"type": "String",
		"description": "The
		alternate AMI to try if
		this first fails".
		
		"default" : "{{amiId}}"
		},
		
		...
		``

Scénario	Commentaire	Exemple

Extension à plusieurs niveaux	Le document définit une variable qui évalue le nom d'une variable. Elle se trouve dans les séparateurs de variables (c'est-à-dire {{ }}) et s'étend à la valeur de cette variable/ce paramètre.	```
		...
		  "parameters": {
		    "firstParameter": {
		      "type": "String",
		      "default": "param2",
		      "description": "The
		 parameter to reference"
		    },
		    "secondParameter": {
		      "type": "String",
		      "default" : "echo
		 {Hello world}",
		      "description": "What
		 to run"
		    }
		  },
		  "mainSteps": [{
		      "name":
		 "runFixedCmds",
		      "action":
		 "aws:runCommand",
		      "maxAttempts": 1,
		      "onFailure":
		 "Continue",
		      "inputs": {
		        "DocumentName":
		 "AWS-RunPowerShellScript",
		
		"InstanceIds" :
		 "{{LaunchInstance.InstanceIds}}",
		        "Parameters": {
		          "commands":
		 ["{{ {{firstParameter}}
		 }}"]
		
		}
		
		...
		
		Note: The customer intention
		 here would be to run a
		 command of "echo {Hello
		 world}"
		``

Scénario	Commentaire	Exemple
Référencement de sortie à partir d'une étape de document qui est un autre type de variable	L'utilisateur fait référence à la sortie d'une étape d'un précédent document au sein d'une étape ultérieure. Le résultat est un type de variable qui ne respecte pas les exigences de l'action dans l'étape suivante.	``` ... mainSteps: - name: getImageId action: aws:executeAwsApi inputs: Service: ec2 Api: DescribeImages Filters: - Name: "name" Values: - "{{ImageName}}" outputs: - Name: ImageIdList Selector: "$.Images" Type: "StringList" - name: copyMyImages action: aws:copyImage maxAttempts: 3 onFailure: Abort inputs: SourceImageId: {{getImageId.ImageIdList}} SourceRegion: ap- northeast-2 ImageName: Encrypted Copies of LAMP base AMI in ap-northeast-2 Encrypted: true ... Note: You must provide the type required by the Automation action. In this case, aws:copyImage requires a "String" type variable but the preceding step outputs a "StringList" type variable. ```

Utilisation des documents Automation

Un document Systems Manager Automation définit les actions exécutées par sur vos instances gérées et autres Systems Manager ressources lors de l'exécution d'une automatisation.AWS Un document contient une ou plusieurs étapes exécutées en ordre séquentiel. Chaque étape est articulée autour d'une seule action. La sortie d'une étape peut être utilisée comme entrée d'une étape ultérieure.

Le processus d'exécution de ces actions et de leurs étapes est appelé flux de travail d'automatisation.

Les types d'action pris en charge pour les documents Automation vous permettent d'automatiser une grande diversité d'opérations dans votre environnement AWS. Par exemple, à l'aide du type `executeScript` d'action, vous pouvez intégrer un python ou un PowerShell script directement dans votre flux de travail. (Lorsque vous créez un document Automation personnalisé, vous pouvez ajouter votre script en ligne ou le joindre à partir d'un compartiment S3 ou de votre ordinateur local.) Vous pouvez automatiser la gestion de vos AWS CloudFormation ressources en utilisant les types d'`createStack`action `deleteStack` et . En outre, grâce au type d'action `executeAwsApi`, une étape peut exécuter n'importe quelle opération API dans n'importe quel service AWS, notamment la création ou la suppression de ressources AWS, le démarrage d'autres processus, le déclenchement de notifications, etc.

Pour obtenir la liste des 20 types d'action pris en charge pour les étapes d'Automation, consultez Référence sur les actions Systems Manager Automation (p. 386).

L'outil AWS Systems Manager Automation offre plusieurs documents Automation comprenant des étapes prédéfinies que vous pouvez utiliser afin d'effectuer des tâches courantes, telles que le redémarrage d'une ou de plusieurs instances EC2 ou la création d'une Amazon Machine Image (AMI). Vous pouvez également créer vos propres documents Automation et les partager avec d'autres comptes AWS ou les rendre publics pour tous les utilisateurs d'Automation.

Les documents Automation sont écrits au moyen JavaScript de JSON (Object Notation) ou YAML. Toutefois, l'outil Document Builder de la console Systems Manager Automation permet de créer un document Automation sans passer par JSON ou YAML.

> **Important**
>
> Si vous exécutez un flux de travail d'automatisation qui appelle d'autres services à l'aide d'un rôle de service AWS Identity and Access Management (IAM), le rôle de service doit être configuré avec l'autorisation d'appeler ces services. Cette exigence s'applique à tous les documents AWS Automation (documents `AWS-*`) comme les documents `AWS-ConfigureS3BucketLogging`, `AWS-CreateDynamoDBBackup` et `AWS-RestartEC2Instance`, pour n'en citer que quelques-uns. Cette exigence s'applique également à tous les documents Automation personnalisés que vous créez qui appellent d'autres services AWS à l'aide d'actions qui appellent d'autres services. Par exemple, si vous utilisez les actions `aws:executeAwsApi`, `aws:createStack` ou `aws:copyImage`, vous devez configurer le rôle de service avec l'autorisation d'appeler ces services. Vous pouvez activer des autorisations à d'autres services AWS en ajoutant une stratégie IAM en ligne au rôle. Pour de plus amples informations, veuillez consulter (Facultatif) Ajoutez une stratégie en ligne Automation pour appeler d'autres services AWS (p. 311).

Pour plus d'informations concernant les actions que vous pouvez spécifier dans un document Systems Manager Automation, consultez Référence sur les actions Systems Manager Automation (p. 386).

Pour plus d'informations sur les documents Automation gérés par AWS qui exécutent des scripts, consultez Documents Automation gérés par Amazon qui exécutent des scripts (p. 458).

Pour plus d'informations sur l'utilisation de l' AWS Toolkit pour Visual Studio Code pour créer des documents Automation, consultez Utilisation des documents Systems Manager Automation dans le Guide de l'utilisateur AWS Toolkit pour Visual Studio Code .

Pour plus d'informations sur l'utilisation de Document Builder pour créer un document Automation personnalisé, consultez Création de documents Automation à l'aide de Document Builder (p. 448).

Pour plus d'informations sur la création de documents Automation personnalisés qui exécutent des scripts, consultez les rubriques suivantes :

- Création de documents Automation qui exécutent des scripts (p. 451) – Fournit des informations sur l'utilisation de Document Builder pour créer un document Automation comprenant l'action `aws:executeScript`.
- Création d'un document Automation exécutant des scripts (ligne de commande) (p. 455) – Fournit des informations sur l'utilisation d'un outil de ligne de commande pour créer un document Automation qui exécute un script.
- Procédure : Utilisation de Document Builder pour créer un document Automation personnalisé (p. 670) – Fournit des conseils étape par étape pour créer un document Automation qui exécute des scripts pour (1) lancer une instance EC2 et (2) attendre que l'état de l'instance devienne `ok`.

Table des matières

- Création de documents Automation à l'aide de Document Builder (p. 448)
- Création d'un document Automation à l'aide de l'éditeur (p. 450)

- Création de documents Automation qui exécutent des scripts (p. 451)
- Création de flux de travail Automation dynamiques avec branchement conditionnel (p. 459)
- Gestion des délais d'expiration dans les documents Automation (p. 469)
- Appel d'autres services AWS à partir d'un flux de travail Systems Manager Automation (p. 470)
- Exemples de scénarios et solutions de documents Automation personnalisées (p. 481)

Création de documents Automation à l'aide de Document Builder

Si les documents Automation AWS Systems Manager publics ne prennent pas en charge toutes les actions que vous souhaitez effectuer sur vos ressources AWS, vous pouvez créer vos propres documents. Pour créer un document Automation personnalisé, vous pouvez créer manuellement un fichier local au format JSON (JavaScriptObject Notation) ou YAML avec les actions d'automatisation appropriées. Sinon, vous pouvez utiliser Document Builder dans la Systems Manager console pour créer plus facilement un document personnalisé.

Grâce à Document Builder, vous pouvez ajouter des étapes d'action Automation à votre document personnalisé et fournir les paramètres requis, sans utiliser la syntaxe JSON ou YAML. Après avoir ajouté des étapes et créé le document, le système convertit les actions que vous avez ajoutées au format YAML que Systems Manager peut utiliser pour exécuter des flux de travail d'automatisation.

Les documents Automation prennent en charge l'utilisation de Markdown, un langage de balisage, qui vous permet d'ajouter des descriptions de style wiki aux documents et des étapes individuelles au sein du document. Pour plus d'informations sur l'utilisation de Markdown, consultez Utilisation de Markdown dans AWS.

> **Tip**
>
> Cette rubrique fournit des informations générales sur l'utilisation de Document Builder avec n'importe quel type d'action pris en charge. Pour plus d'informations sur la création de documents Automation exécutant des scripts, consultez les rubriques suivantes :
>
> - Création de documents Automation qui exécutent des scripts (p. 451) – Fournit des informations sur l'utilisation de Document Builder pour créer un document Automation comprenant l'action `aws:executeScript`.
> - Création d'un document Automation exécutant des scripts (ligne de commande) (p. 455) – Fournit des informations sur l'utilisation d'un outil de ligne de commande pour créer un document Automation qui exécute un script.
> - Procédure : Utilisation de Document Builder pour créer un document Automation personnalisé (p. 670) – Fournit des conseils étape par étape pour créer un document Automation qui exécute des scripts pour (1) lancer une instance Amazon Elastic Compute Cloud (EC2) et (2) attendre que l'état de l'instance devienne `ok`.

Avant de commencer

Avant de créer un document Automation personnalisé à l'aide de Document Builder, nous vous recommandons de prendre connaissance des différentes actions que vous pouvez utiliser dans un document Automation. Pour plus d'informations, consultez Référence sur les actions Systems Manager Automation (p. 386).

Pour créer un document Automation à l'aide de Document Builder

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.

-ou-

Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Documents dans le panneau de navigation.

3. Choisissez Create automation (Créer une automatisation).
4. Dans le champ Name (Nom), tapez un nom descriptif pour le document
5. Pour Document description (Description du document), indiquez la description du style de balisage pour le document. Vous pouvez fournir des instructions sur l'utilisation du document, des étapes numérotées ou tout autre type d'information pour décrire le document. Reportez-vous au texte par défaut pour plus d'informations sur la mise en forme de votre contenu.

   Tip

   Basculez entre Hide preview (Masquer l'aperçu) et Show preview (Afficher l'aperçu) pour voir à quoi ressemble le contenu de votre description lors de la composition.
6. (Facultatif) Pour Assume role (Rôle de responsable), entrez le nom ou l'ARN d'un rôle de service qui effectuera des actions en votre nom. Si vous ne spécifiez pas de rôle, Automation utilise les autorisations d'accès de l'utilisateur qui appelle l'exécution de l'automatisation.

   Important

   Pour les documents Automation qui ne sont pas la propriété d'Amazon et qui utilisent l'action `aws:executeScript`, un rôle doit être spécifié. Pour plus d'informations, consultez la section Autorisations pour exécuter des exécutions Automation (p. 451).
7. (Facultatif) Pour Outputs (Sorties), saisissez toutes les sorties pour l'exécution de ce document à rendre disponibles pour d'autres processus.

   Par exemple, si votre document crée une nouvelle AMI, vous pouvez spécifier ["CreateImage.ImageId"], puis utiliser cette sortie pour créer de nouvelles instances lors d'une exécution d'automatisation ultérieure.
8. (Facultatif) Développez la section Input Parameters (Paramètres d'entrée) et procédez comme suit.

   1. Dans Parameter name (Nom de paramètre), entrez un nom descriptif pour le paramètre de document que vous créez.
   2. Pour Type, choisissez un type pour le paramètre, tel que `String` ou `MapList`.
   3. Pour Required (Obligatoire), effectuez l'une des opérations suivantes :
      - Choisissez Yes (Oui) si une valeur pour ce paramètre de document doit être fournie lors de l'exécution.
      - Choisissez No (Non) si le paramètre n'est pas obligatoire et (facultatif) entrez une valeur de paramètre par défaut dans Defaults value (Valeur par défaut).
   4. Dans Description, entrez une description pour le paramètre de document.

   Note

   Pour ajouter d'autres paramètres de document, choisissez Add a parameter (Ajouter un paramètre). Pour supprimer un paramètre de document, cliquez sur le bouton X (Supprimer).
9. (Facultatif) Développez la section Target type (Type de cible) et choisissez un type de cible pour définir les types de ressources sur lesquelles le document peut s'exécuter. Par exemple, pour exécuter un document sur des instance EC2, choisissez `/AWS::EC2::Instance`.

   Note

   Si vous spécifiez la valeur « `/` », le document peut s'exécuter sur tous les types de ressources. Pour obtenir la liste des types de ressources valides, consultez la Référence des types de ressources AWS dans le AWS CloudFormation Guide de l'utilisateur.

10. (Facultatif) Développez la section Document tags (Balises de document) et entrez une ou plusieurs paires clé-valeur de balise à appliquer au document. Les balises facilitent l'identification, l'organisation et la recherche de ressources. Pour plus d'informations, consultez la section Balisage de documents Systems Manager (p. 1360).
11. Dans la section Step 1 (Étape 1) fournissez les informations suivantes.
    - Pour Step name (Nom de l'étape), entrez un nom descriptif pour la première étape du flux de travail d'automatisation.
    - Pour Action type (Type d'action), sélectionnez le type d'action à utiliser pour cette étape.

      Pour obtenir une liste et des informations sur les types d'action disponibles, consultez Référence sur les actions Systems Manager Automation (p. 386).
    - Dans Description, entrez une description pour l'étape d'automatisation. Vous pouvez utiliser Markdown pour mettre en forme votre texte.
    - Selon le Type d'action sélectionné, saisissez les entrées requises pour le type d'action dans la section Step inputs (Entrées d'étape). Par exemple, si vous avez sélectionné l'action `aws:approve`, vous devez spécifier une valeur pour la propriété `Approvers`.

      Pour plus d'informations sur les champs d'entrée d'étape, consultez l'entrée de Référence sur les actions Systems Manager Automation (p. 386) correspondant au type d'action sélectionné. Par exemple : aws:executeStateMachine – Exécuter une machine d'état AWS Step Functions (p. 421).
    - (Facultatif) Pour Additional inputs (entrées supplémentaires), indiquez toutes les valeurs d'entrée supplémentaires nécessaires à votre document. Les types d'entrée disponibles dépendent du type d'action que vous avez sélectionné pour l'étape. (Notez que certains types d'action nécessitent des valeurs d'entrée.)

      > **Note**
      >
      > Pour ajouter d'autres entrées, choisissez Add optional input (Ajouter une entrée facultative). Pour supprimer une entrée, cliquez sur le bouton X (Supprimer).
    - (Facultatif) Pour Outputs (Sorties), saisissez une sortie pour l'exécution de cette étape à mettre à disposition d'autres processus.

      > **Note**
      >
      > L'option Outputs (Sorties) n'est pas disponible pour tous les types d'action.
    - (Facultatif) Développez la section Common properties (Propriétés communes) et spécifiez les propriétés des actions communes à toutes les actions Automation. Par exemple, pour Timeout seconds (Expiration en secondes), vous pouvez fournir une valeur en secondes pour spécifier la durée de l'exécution de l'étape avant qu'elle ne soit arrêtée.

      Pour plus d'informations, consultez la section Propriétés partagées par toutes les actions (p. 387).

    > **Note**
    >
    > Pour ajouter d'autres étapes de document, sélectionnez Add step (Ajouter étape) et répétez la procédure de création d'une étape. Pour supprimer une étape de document, choisissez Remove step (Supprimer étape).
12. Choisissez Create automation (Créer une automatisation) pour enregistrer le document.

Création d'un document Automation à l'aide de l'éditeur

Si les documents AWS Systems Manager Automation publics n'exécutent pas toutes les actions que vous souhaitez effectuer sur vos ressources AWS, vous pouvez créer vos propres documents. Par exemple, vous pouvez utiliser l'éditeur pour modifier des paramètres, ajouter des étapes supplémentaires à un

document Automation existant ou combiner plusieurs documents Automation en un seul document. Si vous maîtrisez l'écriture de vos propres documents Automation en langage JSON ou YAML, vous pouvez utiliser l'éditeur pour saisir le contenu du document JSON ou YAML.

Pour obtenir des exemples de documents Automation personnalisés, consultez Exemples de scénarios et solutions de documents Automation personnalisées (p. 481).

> **Note**
>
> Si votre document Automation utilise l'action Automation `aws:executeScript` avec le paramètre d'entrée `Attachment`, vous devez utiliser l'AWS CLI ou Document Builder pour créer le document avec succès.

La procédure suivante décrit comment utiliser l'éditeur pour créer un document Automation.

Pour créer un document Automation à l'aide de l'éditeur

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Documents dans le panneau de navigation.
3. Choisissez Create automation (Créer une automatisation).
4. Dans le champ Name (Nom), tapez un nom descriptif pour le document
5. Choisissez l'onglet Editor (Éditeur), puis Edit (Modifier).
6. Saisissez le contenu du document en utilisant le langage JSON ou YAML.
7. Choisissez Create automation (Créer une automatisation).

Création de documents Automation qui exécutent des scripts

Les documents AWS Systems Manager Automation prennent en charge l'exécution de scripts dans le cadre du flux de travail d'automatisation. En utilisant les documents Automation, vous pouvez exécuter des scripts directement dans AWS sans créer un environnement de calcul distinct pour exécuter vos scripts. Comme les documents Automation peuvent exécuter des étapes de script avec d'autres types d'étapes d'automatisation telles que les approbations, vous pouvez intervenir manuellement en cas de situations critiques ou ambiguës.

Autorisations pour exécuter des exécutions Automation

Pour exécuter une exécution d'automatisation, Systems Manager doit utiliser les autorisations d'un rôle AWS Identity and Access Management (IAM). La méthode utilisée par Automation pour déterminer les autorisations du rôle à utiliser dépend de divers facteurs et de si une étape utilise l'action `aws:executeScript`.

Pour les exécutions d'automatisation qui n'utilisent pas `aws:executeScript`, Automation utilise l'une des deux sources d'autorisations suivantes :

- Les autorisations d'un rôle de service IAM, ou rôle de responsable, qui est spécifié dans le document Automation ou transmis en tant que paramètre.
- Si aucun rôle de service IAM n'est spécifié, il utilise les autorisations de l'utilisateur IAM qui a lancé l'exécution de l'automatisation.

Toutefois, lorsqu'une étape d'un document Automation inclut l`aws:executeScript`'action, un rôle de IAM service (rôle de responsable) est toujours requis si le script ou le Python spécifié pour l'action appelle des actions d'API AWS.PowerShell Automation vérifie ce rôle dans l'ordre suivant :

- Les autorisations d'un rôle de service IAM, ou rôle de responsable, qui est spécifié dans le document Automation ou transmis en tant que paramètre.
- Une balise de ressource appliquée au document Automation avec la clé de balise`AutomationScriptExecutionRole`. Dans ce cas, Automation utilise le IAM rôle spécifié en tant que valeur de balise. Par exemple, `arn:aws:iam::123456789012:role/AutomationAssumeRole`.

  Si aucun rôle n'est trouvé, Automation tente d'exécuter le PowerShell script ou Python spécifié pour `aws:executeScript` sans aucune autorisation. Si le script appelle une opération d'API AWS (par exemple l'opération Amazon EC2 `CreateImage`) ou tente d'agir sur une ressource AWS (telle qu'une instance EC2), l'étape contenant le script échoue et Systems Manager renvoie un message d'erreur signalant l'échec.

Pour plus d'informations sur la façon d'exécuter un flux de travail Automation utilisant un rôle de service IAM ou des formes plus avancées d'administration déléguée, consultez Exécution d'une automatisation à l'aide d'un rôle IAM de service (p. 368).

Ajout de scripts aux documents Automation

Vous pouvez ajouter des scripts à vos documents Automation en incluant le script en ligne comme partie d'une étape du document. Vous pouvez également joindre des scripts au document en les téléchargeant de votre ordinateur local ou en spécifiant un compartiment Amazon Simple Storage Service (Amazon S3) où se trouvent les scripts. Lorsqu'une étape exécutant un script est terminée, la sortie du script est disponible en tant qu'objet JSON, que vous pouvez ensuite utiliser comme entrée pour les étapes suivantes de votre flux de travail Automation.

Contraintes de script pour les documents Automation

L'action Automation prend `aws:executeScript` actuellement en charge l'exécution des scripts Python 3.6, Python 3.7 et PowerShell Core 6.0.

Les documents Automation imposent une limite de cinq pièces jointes. Les scripts peuvent être sous la forme d'un script Python (.py), d'un script PowerShell Core (.ps1) ou ils peuvent être attachés en tant que contenu dans un fichier .zip.

Votre compte est facturé pour l'exécution de scripts à l'aide d'Automation. Les étapes d'automatisation qui utilisent l'action `aws:executeScript` sont considérées comme des étapes spéciales. Il n'y a pas de limite d'étape pour les étapes spéciales, mais votre compte est facturé en fonction du nombre d'étapes et de la durée d'exécution de votre script. Pour plus d'informations, consultez la page de Tarification AWS Systems Manager.

Les rubriques suivantes décrivent la création de documents Automation exécutant des scripts.

Rubriques

- Création d'un document Automation exécutant un script (console) (p. 452)
- Création d'un document Automation exécutant des scripts (ligne de commande) (p. 455)
- Documents Automation gérés par Amazon qui exécutent des scripts (p. 458)

Création d'un document Automation exécutant un script (console)

Pour créer un document Automation qui exécute un script

La procédure suivante explique comment utiliser Document Builder dans la console Systems Manager et comment créer un document Automation personnalisé qui exécute un script que vous fournissez.

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Documents dans le panneau de navigation.
3. Choisissez Create automation (Créer une automatisation).
4. Dans le champ Name (Nom), tapez un nom descriptif pour le document
5. Pour Document description (Description du document), indiquez la description du style de balisage pour le document. Vous pouvez fournir des instructions sur l'utilisation du document, des étapes numérotées ou tout autre type d'information pour décrire le document. Reportez-vous au texte par défaut pour plus d'informations sur la mise en forme de votre contenu.

   > **Tip**
   >
   > Basculez entre Hide preview (Masquer l'aperçu) et Show preview (Afficher l'aperçu) pour voir à quoi ressemble le contenu de votre description lors de la composition.
6. (Facultatif) Pour Assume role (Rôle de responsable), entrez le nom ou l'ARN d'un rôle de service qui effectuera des actions en votre nom. Si vous ne spécifiez pas de rôle, Automation utilise les autorisations d'accès de l'utilisateur qui appelle l'exécution de l'automatisation.

   > **Important**
   >
   > Pour les documents Automation qui ne sont pas la propriété d'Amazon et qui utilisent l'action `aws:executeScript`, un rôle doit être spécifié. Pour plus d'informations, consultez la section Autorisations pour exécuter des exécutions Automation (p. 451).
7. (Facultatif) Pour Outputs (Sorties), saisissez toutes les sorties pour l'exécution de ce document à rendre disponibles pour d'autres processus.

   Par exemple, si votre document crée une nouvelle AMI, vous pouvez spécifier ["CreateImage.ImageId"], puis utiliser cette sortie pour créer de nouvelles instances lors d'une exécution d'automatisation ultérieure.
8. (Facultatif) Développez la section Input Parameters (Paramètres d'entrée) et procédez comme suit.

   1. Dans Parameter name (Nom de paramètre), entrez un nom descriptif pour le paramètre de document que vous créez.
   2. Pour Type, choisissez un type pour le paramètre, tel que `String` ou `StringList`.
   3. Pour Required (Obligatoire), effectuez l'une des opérations suivantes.
      - Choisissez Yes (Oui) si une valeur pour ce paramètre de document doit être fournie lors de l'exécution.
      - Choisissez No (Non) si le paramètre n'est pas obligatoire et (facultatif) entrez une valeur de paramètre par défaut dans Defaults value (Valeur par défaut).
   4. Dans Description, entrez une description pour le paramètre de document.

   > **Note**
   >
   > Pour ajouter d'autres paramètres de document, choisissez Add a parameter (Ajouter un paramètre). Pour supprimer un paramètre de document, cliquez sur le bouton X (Supprimer).
9. (Facultatif) Développez les sections Target type (Type de cible) et choisissez un type de cible pour définir les types de ressources sur lesquelles le document peut s'exécuter. Par exemple, pour exécuter un document sur des instance EC2, choisissez `/AWS::EC2::Instance`.

Note

Si vous spécifiez la valeur « `/` », le document peut s'exécuter sur tous les types de ressources. Pour obtenir la liste des types de ressources valides, consultez la Référence des types de ressources AWS dans le AWS CloudFormation Guide de l'utilisateur.

10. Dans la section Step 1 (Étape 1) fournissez les informations suivantes.
    - Pour Step name (Nom de l'étape), entrez un nom descriptif pour la première étape du flux de travail d'automatisation.
    - Pour Action type (Type d'action), sélectionnez Run a script (Exécuter un script) (**`aws:executeScript`**).
    - Dans Description, entrez une description pour l'étape d'automatisation. Vous pouvez utiliser Markdown pour mettre en forme votre texte.
11. Développez la section Inputs (Entrées) et fournissez les informations suivantes.
    - Dans Runtime (Exécution), choisissez le type de script que vous ajoutez. Actuellement, Automation prend en charge Python 3.6, Python 3.7 et PowerShell Core 6.0.
    - Pour Handler (Gestionnaire), saisissez l'entrée pour l'exécution du script, généralement le nom de la fonction de votre script. (Non requis pour PowerShell .)

      Important

      Vous devez vous assurer que la fonction définie dans le gestionnaire possède deux paramètres, `events` et `context` . Par exemple, vous devez entrer `launch_instance` si votre script a commencé par ce qui suit.

      ```
      def launch_instance(events, context):
        import boto3
        ec2 = boto3.client('ec2')

      [...truncated...]
      ```

    - Pour Script, choisissez une méthode pour fournir un script au document Automation.
      - Pour incorporer un script dans le document, entrez le code de script dans la zone de texte.

        -ou-

      - Pour Attachment (Pièce jointe), choisissez Stored on my machine (Enregistré sur mon ordinateur) ou Upload S3 File URL (Télécharger l'URL d'un fichier S3).

        Si vous choisissez Stocké sur mon ordinateur : Pour Amazon S3 URL, entrez l'emplacement d'un compartiment S3 dans votre compte où vous souhaitez stocker la pièce jointe de chargement, puis choisissez Upload pour rechercher et sélectionner le fichier.

        Si vous choisissez Upload S3 File URL (Télécharger l'URL d'un fichier S3), fournissez les informations suivantes :
        - URL du fichier S3 : Saisissez l'emplacement dans un compartiment S3 de votre compte où le fichier est stocké.
        - Nom de fichier : Saisissez le nom du fichier.
        - Total de contrôle du fichier : Saisissez le total de contrôle du fichier à l'aide de l'algorithme sha256.

          Tip

          Vous pouvez calculer le total de contrôle du fichier dans sha256 en utilisant un outil tel que shasum sous Linux. Par exemple : shasum -a 256 /path/to/file. Dans Windows, vous pouvez utiliser l'`Get-FileHash`PowerShellapplet de commande pour obtenir les

mêmes informations. Notez que le total de contrôle ETag ou md5 ne fonctionnera pas pour cette valeur.

12. (Facultatif) Développez Additional inputs (Entrées supplémentaires) et procédez comme suit.
    - Pour Input name (Nom d'entrée), choisissez `InputPayload`. - Entrée de fonction au format YAML.
    - Pour Input value (Valeur d'entrée), saisissez votre entrée de script au format YAML.
13. (Facultatif) Développez Outputs (Sorties) et entrez les valeurs de Name (Nom), Selector (Sélecteur) et Type de toutes les sorties à créer à partir de cette étape. La sortie de l'étape peut être utilisée dans les étapes suivantes de votre document Automation. Voici quelques exemples.

    Nom : **`myInstance`**| Sélecteur : **`$.InstanceInformationList[0].InstanceId`**| Type :**`String`**

    Nom : `platform`| Sélecteur : `$.Reservations[0].Instances[0].Platform`| Type :**`String`**

    Nom : `message`| Sélecteur : $.Payload.message | Type :**`String`**

    Pour plus d'informations sur les sorties, consultez Utilisation des entrées et des sorties (p. 473).

    > **Tip**
    >
    > Pour ajouter d'autres sorties, sélectionnez Add output (Ajouter une sortie).

14. (Facultatif) Développez la section Common properties (Propriétés communes) et spécifiez les propriétés des actions communes à toutes les actions Automation. Par exemple, pour Timeout seconds (Expiration en secondes), vous pouvez fournir une valeur en secondes pour spécifier la durée de l'exécution de l'étape avant qu'elle ne soit arrêtée.

    Pour plus d'informations, consultez la section Propriétés partagées par toutes les actions (p. 387).

    > **Note**
    >
    > Pour ajouter d'autres étapes de document, sélectionnez Add step (Ajouter étape) et répétez la procédure de création d'une étape. Pour supprimer une étape de document, choisissez Remove step (Supprimer étape).

15. Choisissez Create document (Créer un document) pour enregistrer le document.

Création d'un document Automation exécutant des scripts (ligne de commande)

Les exemples suivants montrent comment utiliser l'AWS CLI (sous Linux ou Windows Server) ou Outils AWS pour PowerShell pour créer un document Automation qui exécute un script à l'aide du paramètre `Attachment`.

Avant de commencer

Avant de commencer, assurez-vous d'avoir préparé les ressources suivantes.

- Le contenu de votre document Automation dans un fichier au format YAML avec une extension .yaml, par exemple, `MyAutomationDocument.yaml`.
- Les fichiers à joindre au document. Vous pouvez joindre des fichiers de script ou des fichiers .zip.

  Pour les scripts, Automation prend en charge Python 3.6 et 3.7, PowerShell Core 6.0.
- Installez et configurez l'AWS CLI ou Outils AWS pour PowerShell si vous ne l'avez pas déjà fait.

  Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

Joindre un seul fichier à partir d'un compartiment S3

Exécutez la commande suivante pour créer un document Automation à l'aide d'un fichier de script stocké dans un compartiment S3.

Linux

```
aws ssm create-document \
  --name CustomAutomationScript \
  --content file://AutomationDocument.yaml \
  --document-format YAML \
  --attachments "Key=S3FileUrl,Name=script.py,Values=https://DOC-EXAMPLE-BUCKET.s3-aws-region.amazonaws.com/filePath" \
  --document-type Automation
```

Windows

```
aws ssm create-document ^
  --name CustomAutomationScript ^
  --content file://AutomationDocument.yaml ^
  --document-format YAML ^
  --attachments Key=S3FileUrl,Name=script.py,Values="https://DOC-EXAMPLE-BUCKET.s3-aws-region.amazonaws.com/filePath" ^
  --document-type Automation
```

PowerShell

```
New-SSMDocument `
  -Name CustomAutomationScript `
  -Content file://AutomationDocument.yaml `
  -DocumentFormat YAML `
  -Attachments @{
      "Key"="S3FileUrl";
      "Name"="script.py";
      "Values"="https://DOC-EXAMPLE-BUCKET.s3-aws-region.amazonaws.com/filePath"
    }, @{
      "Key"="S3FileUrl";
      "Name"="Zip-file.zip";
      "Values"="https://DOC-EXAMPLE-BUCKET.s3-aws-region.amazonaws.com/filePath"
    } `
  -DocumentType Automation
```

Joindre plusieurs fichiers à partir d'un compartiment S3

Exécutez la commande suivante pour créer un document Automation à l'aide d'un script ou de plusieurs fichiers de script stockés dans un compartiment S3. Notez qu'une clé `Name` pour les fichiers n'est pas spécifiée. La commande joint tous les fichiers pris en charge à partir de l'emplacement du compartiment S3.

Linux

```
aws ssm create-document \
  --name CustomAutomationScript \
  --content file://AutomationDocument.yaml \
  --document-format YAML \
  --attachments Key=SourceUrl,Values="https://DOC-EXAMPLE-BUCKET.s3-aws-region.amazonaws.com/filePath" \
  --document-type Automation
```

Windows

```
aws ssm create-document ^
```

```
  --name CustomAutomationScript ^
  --content file://AutomationDocument.yaml ^
  --document-format YAML ^
  --attachments Key=SourceUrl,Values="https://DOC-EXAMPLE-BUCKET.s3-aws-
region.amazonaws.com/filePath" ^
  --document-type Automation
```

PowerShell

```
New-SSMDocument `
  -Name CustomAutomationScript `
  -Content file://AutomationDocument.yaml `
  -DocumentFormat YAML `
  -Attachments @{
      "Key"="SourceUrl";
      "Values"="https://DOC-EXAMPLE-BUCKET.s3-aws-region.amazonaws.com/filePath"
    } `
  -DocumentType Automation
```

Joindre des fichiers à partir d'un autre document Automation de votre compte AWS

Exécutez la commande suivante pour créer un document Automation à l'aide d'un script déjà attaché à un autre document Automation de votre compte.

Le format de la valeur de la clé dans cette commande est *document-name/document-version-number/file-name*Par exemple :

```
"MyDocument/2/script.py"
```

Linux

```
aws ssm create-document \
  --name CustomAutomationScript \
  --content file://AutomationDocument.yaml \
  --document-format YAML \
  --attachments Key=AttachmentReference,Values="document-name/document-version-number/
file-name" \
  --document-type Automation
```

Windows

```
aws ssm create-document ^
  --name CustomAutomationScript ^
  --content file://AutomationDocument.yaml ^
  --document-format YAML ^
  --attachments Key=AttachmentReference,Values="document-name/document-version-number/
file-name" ^
  --document-type Automation
```

PowerShell

```
New-SSMDocument `
  -Name CustomAutomationScript `
  -Content file://AutomationDocument.yaml `
  -DocumentFormat YAML `
  -Attachments @{
      "Key"="AttachmentReference";
      "Values"="document-name/document-version-number/file-name"
    } `
```

```
  -DocumentType Automation
```

Joindre des fichiers à partir d'un document Automation d'un autre compte AWS

Exécutez la commande suivante pour créer un document Automation à l'aide d'un script déjà attaché à un document Automation qui a été partagé avec vous à partir d'un autre compte AWS.

Le format de la valeur de la clé dans cette commande est *document-arn/document-version-number/file-name*Par exemple :

```
"arn:aws:ssm:us-east-2:123456789012:document/OtherAccountDocument/2/script.py"
```

Linux

```
aws ssm create-document \
  --name CustomAutomationScript \
  --content file://AutomationDocument.yaml \
  --document-format YAML \
  --attachments Key=AttachmentReference,Values="document-arn/document-version-number/
file-name/file-name" \
  --document-type Automation
```

Windows

```
aws ssm create-document ^
  --name CustomAutomationScript ^
  --content file://AutomationDocument.yaml ^
  --document-format YAML ^
  --attachments Key=AttachmentReference,Values="document-arn/document-version-number/
file-name" ^
  --document-type Automation
```

PowerShell

```
New-SSMDocument `
  -Name CustomAutomationScript `
  -Content file://AutomationDocument.yaml `
  -DocumentFormat YAML `
  -Attachments @{
      "Key"="AttachmentReference";
      "Values"="document-arn/document-version-number/file-name"
    } `
  -DocumentType Automation
```

Documents Automation gérés par Amazon qui exécutent des scripts

Les documents AWS Systems Manager Automation prennent en charge l'exécution de scripts dans le cadre de l'exécution de l'automatisation.

Les documents Automation suivants gérés par AWS comprennent la prise en charge de l'exécution de scripts.

> **Note**
>
> Vous pouvez également créer vos propres documents Automation personnalisés qui peuvent exécuter des scripts. Pour plus d'informations, consultez la section Création de documents Automation qui exécutent des scripts (p. 451).

- AWS-CreateRdsSnapshot (p. 544) – Crée un instantané Amazon Relational Database Service (Amazon RDS) pour une instance Amazon RDS.
- AWS-CreateServiceNowIncident (p. 545)–Crée un incident dans la table des ServiceNow incidents.
- AWS-ExportOpsDataToS3 (p. 560)–Récupère une liste de OpsData résumés dans AWS Systems Manager l'Explorateur et les exporte vers un objet dans un compartiment S3 spécifié.
- AWS-RunCfnLint (p. 577)–Utilise un AWS CloudFormation Linter (cfn-python-lint) pour valider les modèles YAML et JSON par rapport à la spécification de AWS CloudFormation ressource. Ce document Automation utilise cfn-lint v0.24.4.
- AWS-RunPacker (p. 579)–Utilise l'HashiCorpoutil Packertool pour valider, corriger ou créer des modèles de packer utilisés pour créer des images de machine. Ce document utilise Packer v1.4.4.

Création de flux de travail Automation dynamiques avec branchement conditionnel

Par défaut, les étapes que vous définissez dans la section `mainSteps` d'un document Automation sont exécutées par ordre séquentiel. Lorsqu'une action est terminée, la prochaine action spécifiée dans la section `mainSteps` commence. En outre, si l'exécution d'une action échoue, l'intégralité du flux de travail Automation échoue (par défaut). Vous pouvez utiliser les options d'action Automation `aws:branch` et de document d'Automation décrites dans cette section pour créer des flux de travail Automation qui effectuent des ramifications conditionnelles. Autrement dit, vous pouvez créer des flux de travail Automation qui passent à une autre étape après avoir évalué différentes possibilités ou qui répondent aux modifications de manière dynamique à la fin d'une étape. Voici la liste des options que vous pouvez utiliser pour créer des flux de travail Automation dynamiques :

- aws:branch : Cette action d'automatisation vous permet de créer un flux de travail Automation dynamique qui évalue plusieurs options en une seule étape, puis passe à une autre étape du document Automation en fonction des résultats de cette évaluation.
- nextStep: Cette option spécifie l'étape du flux de travail Automation à traiter immédiatement après la fin d'une étape.
- isEnd: Cette option arrête une exécution d'Automation à la fin d'une étape spécifique. La valeur par défaut de cette option est false.
- isCritical: Cette option désigne une étape comme étant critique pour la réussite de l'exécution d'Automation. Si une étape portant cette désignation échoue, Automation signale l'état final de l'automatisation comme Échouée. La valeur par défaut de cette option est true.
- onFailure: Cette option indique si le flux de travail doit être abandonné, continuer ou passer à une autre étape en cas d'échec. La valeur par défaut de cette option est Abort.

La section suivante décrit l'action Automation `aws:branch`. Pour de plus amples informations sur les options de flux de travail `nextStep`, `isEnd`, `isCritical` et `onFailure`, veuillez consulter Exemples d'utilisation des options de flux de travail dynamique (p. 467).

Utilisation de l'action aws:branch

L'action `aws:branch` propose les options les plus dynamiques des ramifications conditionnelles pour les flux de travail Automation. Comme noté précédemment, cette action permet à vos flux de travail Automation d'évaluer plusieurs conditions en une seule étape, puis de passer à une nouvelle étape en fonction des résultats de cette évaluation. L'action `aws:branch` fonctionne comme une instruction de programmation `IF-ELIF-ELSE`.

Voici un exemple YAML d'étape `aws:branch` :

```
- name: ChooseOSforCommands
```

```
  action: aws:branch
  inputs:
    Choices:
    - NextStep: runPowerShellCommand
      Variable: "{{GetInstance.platform}}"
      StringEquals: Windows
    - NextStep: runShellCommand
      Variable: "{{GetInstance.platform}}"
      StringEquals: Linux
    Default:
      PostProcessing
```

Lorsque vous spécifiez l'action `aws:branch` pour une étape, vous définissez des `Choices` que le flux de travail doit évaluer. Les flux de travail peuvent évaluer ces `Choices` en fonction de la valeur d'un paramètre que vous avez spécifiée dans la section `Parameters` du document d'Automation. Les flux de travail peuvent également évaluer ces `Choices` en fonction de la sortie d'une étape précédente.

Le flux de travail d'automatisation évalue chaque choix à l'aide d'une expression booléenne. Si l'évaluation détermine que le premier choix est true, le flux de travail accède à l'étape désignée pour ce choix. Si l'évaluation détermine que le premier choix est false, le flux de travail évalue le choix suivant. Si l'étape inclut au moins trois `Choices`, le flux de travail évalue chaque choix par ordre séquentiel jusqu'à ce qu'il en identifie un possédant la valeur true. Le flux de travail accède ensuite à l'étape désignée correspondant au choix défini sur true.

Si aucun des `Choices` n'est true, le flux de travail vérifie si l'étape contient une valeur `Default`. La valeur `Default` définit une étape à laquelle le flux de travail doit passer si aucun des choix n'est défini sur true. Si aucune valeur `Default` n'est spécifiée pour l'étape, le flux de travail d'automatisation traite l'étape suivante du document.

Voici une `aws:branch` étape au format YAML nommée chooseOSfromParameter . Cette étape comprend deux `Choices` : (`NextStep: runWindowsCommand`) et (`NextStep: runLinuxCommand`). Le flux de travail Automation évalue ces `Choices` pour déterminer quelle commande exécuter pour le système d'exploitation approprié. La `Variable` pour chaque choix utilise `{{OSName}}`, qui est un paramètre défini par l'auteur dans la section `Parameters` du document.

```
mainSteps:
- name: chooseOSfromParameter
  action: aws:branch
  inputs:
    Choices:
    - NextStep: runWindowsCommand
      Variable: "{{OSName}}"
      StringEquals: Windows
    - NextStep: runLinuxCommand
      Variable: "{{OSName}}"
      StringEquals: Linux
```

Voici une `aws:branch` étape au format YAML nommée chooseOSfromOutput . Cette étape comprend deux `Choices` : (`NextStep: runPowerShellCommand`) et (`NextStep: runShellCommand`). Le flux de travail Automation évalue ces `Choices` pour déterminer quelle commande exécuter pour le système d'exploitation approprié. La `Variable` pour chaque choix utilise `{{GetInstance.platform}}`, qui est la sortie d'une étape antérieure dans le document. Cet exemple inclut également une option appelée `Default` . Si le flux de travail évalue les deux `Choices` et qu'aucun des deux n'a la valeur true, le flux de travail Automation accède à une étape appelée `PostProcessing` .

```
mainSteps:
- name: chooseOSfromOutput
  action: aws:branch
```

```
inputs:
  Choices:
  - NextStep: runPowerShellCommand
    Variable: "{{GetInstance.platform}}"
    StringEquals: Windows
  - NextStep: runShellCommand
    Variable: "{{GetInstance.platform}}"
    StringEquals: Linux
  Default:
    PostProcessing
```

Création d'une étape `aws:branch` dans un document d'Automation

Lorsque vous créez une étape `aws:branch` dans un document d'Automation, vous définissez les `Choices` que le flux de travail doit évaluer pour déterminer l'étape suivante à laquelle il doit passer. Comme indiqué précédemment, l'évaluation des `Choices` repose sur une expression booléenne. Chaque choix doit définir les options suivantes :

- NextStep: Étape suivante à traiter dans le document Automation si le choix désigné est true.
- Variable : Spécifiez le nom d'un paramètre qui est défini dans la `Parameters` section du document d'Automation ou spécifiez un objet de sortie d'une étape précédente dans le document d'Automation.

  Spécifiez les variables de paramètre des variables sous la forme suivante :

  `Variable: "{{name_of_parameter}}"`

  Spécifiez les variables d'objet de sortie sous la forme suivante :

  `Variable: "{{previousStepName.outputFieldName}}"`

  > **Note**
  >
  > La création de la variable de sortie est décrite plus en détail dans la section suivante, À propos de la création de la variable de sortie (p. 463).

- Opération : Critères utilisés pour évaluer le choix, par exemple `StringEquals: Linux` . `Laws:branch`'action prend en charge les opérations suivantes :

  ### Opérations de chaîne

  - StringEquals
  - EqualsIgnoreCase
  - StartsWith
  - EndsWith
  - Contains

  ### Opérations numériques

  - NumericEquals
  - NumericGreater
  - NumericLesser
  - NumericGreaterOrEquals
  - NumericLesser
  - NumericLesserOrEquals

  ### Opération booléenne

  - BooleanEquals

> **Important**
>
> Lorsque vous créez un document d'Automation, le système vérifie chaque opération dans le document. Si une opération n'est pas prise en charge, le système renvoie une erreur lorsque vous tentez de créer le document.

- Par défaut : Spécifiez une étape de secours à laquelle le flux de travail doit passer si aucune des n'`Choices`est true.

> **Note**
>
> Si vous ne voulez pas spécifier de valeur `Default`, vous pouvez spécifier le flux de travail `isEnd`. Si aucun des `Choices` n'est true et qu'aucune valeur `Default` n'est spécifiée, le flux de travail Automation s'arrête à la fin de l'étape.

Utilisez les modèles suivants pour vous aider à construire l'étape `aws:branch` dans vos documents Automation :

YAML

```
mainSteps:
- name: a name for the step
  action: aws:branch
  inputs:
    Choices:
    - NextStep: step to jump to if evaluation for this choice is true
      Variable: "{{parameter name or output from previous step}}"
      Operation type: Operation value
    - NextStep: step to jump to if evaluation = true
      Variable: "{{parameter name or output from previous step}}"
      Operation type: Operation value
    Default:
      step to jump to if all choices are false
```

JSON

```
{
   "mainSteps":[
      {
         "name":"a name for the step",
         "action":"aws:branch",
         "inputs":{
            "Choices":[
               {
                  "NextStep":"step to jump to if evaluation for this choice is true",
                  "Variable":"{{parameter name or output from previous step}}",
                  "Operation type":"Operation value"
               },
               {
                  "NextStep":"step to jump to if evaluation = true",
                  "Variable":"{{parameter name or output from previous step}}",
                  "Operation type":"Operation value"
               }
            ],
            "Default":"step to jump to if all choices are false"
         }
      }
   ]
}
```

À propos de la création de la variable de sortie

Pour créer un choix `aws:branch` qui fait référence à la sortie d'une étape précédente, vous devez identifier le nom de l'étape précédente et le nom du champ de sortie. Vous devez ensuite combiner les noms de l'étape et du champ en utilisant le format suivant :

`Variable: "{{previousStepName.outputFieldName}}"`

Par exemple, la première étape ci-dessous se nomme `GetInstance` . Puis, sous `outputs` , se trouve un champ appelé `platform` . Dans la deuxième étape (`ChooseOSforCommands`), l'auteur souhaite référencer la sortie du champ de plateforme en tant que variable. Pour créer la variable, il vous suffit de combiner le nom de l'étape (GetInstance) et le nom du champ de sortie (plateforme) pour créer `Variable: "{{GetInstance.platform}}"`.

```
mainSteps:
- Name: GetInstance
  action: aws:executeAwsApi
  inputs:
    Service: ssm
    Api: DescribeInstanceInformation
    Filters:
    - Key: InstanceIds
      Values: ["{{ InstanceId }}"]
  outputs:
  - Name: myInstance
    Selector: "$.InstanceInformationList[0].InstanceId"
    Type: String
  - Name: platform
    Selector: "$.InstanceInformationList[0].PlatformType"
    Type: String
- name: ChooseOSforCommands
  action: aws:branch
  inputs:
    Choices:
    - NextStep: runPowerShellCommand
      Variable: "{{GetInstance.platform}}"
      StringEquals: Windows
    - NextStep: runShellCommand
      Variable: "{{GetInstance.platform}}"
      StringEquals: Linux
    Default:
      Sleep
```

Voici un exemple JSON qui montre comment `"Variable": "{{ describeInstance.Platform }}"` est créé à partir de l'étape précédente (`"describeInstance"`) et le champ de sortie (`"Platform"`).

```
{
      "name": "describeInstance",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "DescribeInstances",
        "InstanceIds": [
          "{{ InstanceId }}"
        ]
      },
      "outputs": [
        {
          "Name": "Platform",
          "Selector": "$.Reservations[0].Instances[0].Platform",
          "Type": "String"
```

```
        }
      ],
      "nextStep": "branchOnInstancePlatform"
    },
    {
      "name": "branchOnInstancePlatform",
      "action": "aws:branch",
      "inputs": {
        "Choices": [
          {
            "NextStep": "runEC2RescueForWindows",
            "Variable": "{{ describeInstance.Platform }}",
            "StringEquals": "windows"
          }
        ],
        "Default": "runEC2RescueForLinux"
      }
    }
```

Exemple de documents Automation aws:branch

Voici quelques exemples de documents Automation qui utilisent `aws:branch`.

Exemple 1 : Utilisation d'aws:branch avec une variable de sortie pour exécuter des commandes en fonction du type de système d'exploitation

Dans la première étape de cet exemple de document (`GetInstance`), l'auteur utilise l' action `aws:executeAwsApi` pour appeler l'action d'API `ssm` `DescribeInstanceInformation`. L'auteur utilise cette action pour déterminer le type de système d'exploitation utilisé par une instance. L'action `aws:executeAwsApi` génère l'ID d'instance et le type de plateforme.

Dans la deuxième étape (`ChooseOSforCommands`), l'auteur utilise l'action `aws:branch` avec deux `Choices` (`NextStep: runPowerShellCommand`) et (`NextStep: runShellCommand`). Le flux de travail Automation évalue le système d'exploitation de l'instance en utilisant la sortie de l'étape précédente (`Variable: "{{GetInstance.platform}}"`). Le flux de travail Automation accède à une étape pour le système d'exploitation désigné.

```
---
schemaVersion: '0.3'
assumeRole: "{{AutomationAssumeRole}}"
parameters:
  AutomationAssumeRole:
    default: ""
    type: String
mainSteps:
- name: GetInstance
  action: aws:executeAwsApi
  inputs:
    Service: ssm
    Api: DescribeInstanceInformation
  outputs:
  - Name: myInstance
    Selector: "$.InstanceInformationList[0].InstanceId"
    Type: String
  - Name: platform
    Selector: "$.InstanceInformationList[0].PlatformType"
    Type: String
- name: ChooseOSforCommands
  action: aws:branch
  inputs:
    Choices:
    - NextStep: runPowerShellCommand
      Variable: "{{GetInstance.platform}}"
```

```
      StringEquals: Windows
    - NextStep: runShellCommand
      Variable: "{{GetInstance.platform}}"
      StringEquals: Linux
    Default:
      Sleep
- name: runShellCommand
  action: aws:runCommand
  inputs:
    DocumentName: AWS-RunShellScript
    InstanceIds:
    - "{{GetInstance.myInstance}}"
    Parameters:
      commands:
      - ls
  isEnd: true
- name: runPowerShellCommand
  action: aws:runCommand
  inputs:
    DocumentName: AWS-RunPowerShellScript
    InstanceIds:
    - "{{GetInstance.myInstance}}"
    Parameters:
      commands:
      - ls
  isEnd: true
- name: Sleep
  action: aws:sleep
  inputs:
    Duration: PT3S
```

Exemple 2 : Utilisation d'aws:branch avec une variable de paramètre pour exécuter des commandes en fonction du type de système d'exploitation

L'auteur définit plusieurs options de paramètres au début du document dans la section `parameters`. Un paramètre est nommé `OperatingSystemName` . Dans la première étape (`ChooseOS`), l'auteur utilise l'`aws:branch`action avec deux `Choices` (`NextStep: runWindowsCommand`) et (`NextStep: runLinuxCommand`). La variable pour ces `Choices` référence l'option de paramètre spécifiée dans la section Paramètres (`Variable: "{{OperatingSystemName}}"`). Lorsque l'utilisateur exécute ce flux de travail Automation, il spécifie une valeur au moment de l'exécution pour `OperatingSystemName` . Le flux de travail Automation utilise le paramètre d'exécution lors de l'`Choices`évaluation. Le flux de travail Automation accède à une étape pour le système d'exploitation désigné en fonction du paramètre d'exécution spécifié pour `OperatingSystemName`.

```
---
schemaVersion: '0.3'
assumeRole: "{{AutomationAssumeRole}}"
parameters:
  AutomationAssumeRole:
    default: ""
    type: String
  OperatingSystemName:
    type: String
  LinuxInstanceId:
    type: String
  WindowsInstanceId:
    type: String
mainSteps:
- name: ChooseOS
  action: aws:branch
  inputs:
    Choices:
    - NextStep: runWindowsCommand
```

```
      Variable: "{{OperatingSystemName}}"
      StringEquals: windows
    - NextStep: runLinuxCommand
      Variable: "{{OperatingSystemName}}"
      StringEquals: linux
    Default:
      Sleep
- name: runLinuxCommand
  action: aws:runCommand
  inputs:
    DocumentName: "AWS-RunShellScript"
    InstanceIds:
    - "{{LinuxInstanceId}}"
    Parameters:
      commands:
      - ls
  isEnd: true
- name: runWindowsCommand
  action: aws:runCommand
  inputs:
    DocumentName: "AWS-RunPowerShellScript"
    InstanceIds:
    - "{{WindowsInstanceId}}"
    Parameters:
      commands:
      - date
  isEnd: true
- name: Sleep
  action: aws:sleep
  inputs:
    Duration: PT3S
```

Création de documents à ramifications complexes avec des opérateurs

Vous pouvez créer des documents à ramifications complexes avec les opérateurs `And`, `Or` et `Not` dans les étapes `aws:branch`.

L'opérateur AND

Utilisez l'opérateur `And` lorsque vous voulez que plusieurs variables soient true pour un choix. Dans l'exemple suivant, le premier choix évalue si une instance est `running` et utilise le système d'exploitation `Windows`. Si l'évaluation de ces deux variables a la valeur true, le flux de travail Automation accède à l'étape `runPowerShellCommand`. Si une ou plusieurs des variables sont false, le flux de travail évalue les variables du deuxième choix.

```
mainSteps:
- name: switch2
  action: aws:branch
  inputs:
    Choices:
    - And:
      - Variable: "{{GetInstance.pingStatus}}"
        StringEquals: running
      - Variable: "{{GetInstance.platform}}"
        StringEquals: Windows
      NextStep: runPowerShellCommand

    - And:
      - Variable: "{{GetInstance.pingStatus}}"
        StringEquals: running
      - Variable: "{{GetInstance.platform}}"
        StringEquals: Linux
      NextStep: runShellCommand
    Default:
```

```
      sleep3
```

L'opérateur OR

Utilisez l'opérateur `Or` lorsque vous voulez qu'au moins une des variables soient true pour un choix. Dans l'exemple suivant, le premier choix évalue si un paramètre de chaîne est `Windows` et si la sortie d'une étape AWS Lambda est true. Si l'évaluation détermine que l'une de ces variables a la valeur true, le flux de travail Automation accède à l'étape `RunPowerShellCommand`. Si les deux variables sont false, le flux de travail évalue les variables du deuxième choix.

```
- Or:
  - Variable: "{{parameter1}}"
    StringEquals: Windows
  - Variable: "{{BooleanParam1}}"
    BooleanEquals: true
  NextStep: RunPowershellCommand
- Or:
  - Variable: "{{parameter2}}"
    StringEquals: Linux
  - Variable: "{{BooleanParam2}}"
    BooleanEquals: true
  NextStep: RunShellScript
```

L'opérateur NOT

Utilisez l'opérateur `Not` lorsque vous souhaitez accéder à une étape définie lorsqu'une variable n'est pas true. Dans l'exemple suivant, le premier choix évalue si une chaîne de paramètre est `Not Linux` . Si l'évaluation détermine que la variable n'est pas Linux, le flux de travail Automation accède à l'`sleep2`étape. Si l'évaluation du premier choix détermine que la variable est Linux, le flux de travail évalue le choix suivant.

```
mainSteps:
- name: switch
  action: aws:branch
  inputs:
    Choices:
    - NextStep: sleep2
      Not:
        Variable: "{{testParam}}"
        StringEquals: Linux
    - NextStep: sleep1
      Variable: "{{testParam}}"
      StringEquals: Windows
    Default:
      sleep3
```

Exemples d'utilisation des options de flux de travail dynamique

Cette section inclut différents exemples d'utilisation des options de flux de travail dynamique dans un document d'Automation. Chaque exemple présenté dans cette section étend le document d'Automation suivant. Ce document a deux actions. La première action se nomme `InstallMsiPackage` . Elle utilise l'`aws:runCommand`action pour installer une application sur une Windows Server instance. La deuxième action se nomme `TestInstall` . Elle utilise l'`aws:invokeLambdaFunction`action pour effectuer un test de l'application installée si celle-ci a été installée correctement. . L'étape 1 spécifie `onFailure: Abort` . Cela signifie que si l'application n'a pas été installée avec succès, l'exécution du flux de travail Automation s'arrête avant la deuxième étape.

Exemple 1 : Document Automation avec deux actions linéaires

```
---
```

```
schemaVersion: '0.3'
description: Install MSI package and run validation.
assumeRole: "{{automationAssumeRole}}"
parameters:
  automationAssumeRole:
    type: String
    description: "(Required) Assume role."
  packageName:
    type: String
    description: "(Required) MSI package to be installed."
  instanceIds:
    type: String
    description: "(Required) Comma separated list of instances."
mainSteps:
- name: InstallMsiPackage
  action: aws:runCommand
  maxAttempts: 2
  onFailure: Abort
  inputs:
    InstanceIds:
    - i-02573cafcfEXAMPLE
    - i-0471e04240EXAMPLE
    - i-07782c72faEXAMPLE
    DocumentName: AWS-RunPowerShellScript
    Parameters:
      commands:
      - msiexec /i {{packageName}}
- name: TestInstall
  action: aws:invokeLambdaFunction
  maxAttempts: 1
  timeoutSeconds: 500
  inputs:
    FunctionName: TestLambdaFunction
...
```

Création d'un flux de travail dynamique qui passe à différentes étapes à l'aide de l'onFailureoption

L'exemple suivant utilise les options `onFailure: step:`*`step_name`*, `nextStep` et `isEnd` pour créer un flux de travail Automation dynamique. Avec cet exemple, si l'`InstallMsiPackage`action échoue, le flux de travail accède à une action appelée PostFailure (`onFailure: step:PostFailure`) pour exécuter une AWS Lambda fonction afin d'effectuer une action au cas où l'installation échoue. Si l'installation réussit, le processus de flux de travail accède à l'TestInstallaction (`nextStep: TestInstall`). Les étapes `TestInstall` et `PostFailure` utilisent l'option `isEnd` (`isEnd: true`) pour que le flux de travail termine l'exécution du flux de travail lorsque l'une de ces étapes est terminée.

> **Note**
>
> L'utilisation de l'option `isEnd` à la dernière étape de la section `mainSteps` est facultative. Si la dernière étape ne fait pas passer à d'autres étapes, le flux de travail Automation s'arrête après l'exécution de l'action effectuée à la dernière étape.

Exemple 2 : Un flux de travail dynamique qui accède à différentes étapes

```
mainSteps
- name: InstallMsiPackage
  action: aws:runCommand
  onFailure: step:PostFailure
  maxAttempts: 2
  inputs:
    InstanceIds:
    - i-02573cafcfEXAMPLE
    - i-0471e04240EXAMPLE
    DocumentName: AWS-RunPowerShellScript
    Parameters:
```

```
      commands:
      - msiexec /i {{packageName}}
  nextStep: TestInstall
- name: TestInstall
  action: aws:invokeLambdaFunction
  maxAttempts: 1
  timeoutSeconds: 500
  inputs:
    FunctionName: TestLambdaFunction
  isEnd: true
- name: PostFailure
  action: aws:invokeLambdaFunction
  maxAttempts: 1
  timeoutSeconds: 500
  inputs:
    FunctionName: PostFailureRecoveryLambdaFunction
  isEnd: true
...
```

> **Note**
>
> Avant de traiter un document d'Automation, le système vérifie que le document ne crée pas une boucle infinie. Si une boucle infinie est détectée, Automation renvoie une erreur, avec un cercle qui entoure les étapes qui créent la boucle.

Création d'un flux de travail dynamique qui définit des étapes critiques

Vous pouvez spécifier qu'une étape est critique pour la réussite globale du flux de travail Automation. Si une étape critique échoue, Automation signale l'état comme ayant échoué, même si une ou plusieurs étapes se sont déroulées correctement. Dans l'exemple suivant, l'utilisateur identifie l'VerifyDependenciesétape si l'InstallMsiPackageétape échoue (`onFailure: step:VerifyDependencies`). L'utilisateur spécifie que l'étape `InstallMsiPackage` n'est pas critique (`isCritical: false`). Dans cet exemple, si l'installation de l'application a échoué, Automation traite l'étape `VerifyDependencies` pour déterminer si une ou plusieurs dépendances sont manquantes, ce qui aurait provoqué l'échec de l'installation.

Exemple 3 : Définition des étapes critiques pour le flux de travail Automation

```
---
name: InstallMsiPackage
action: aws:runCommand
onFailure: step:VerifyDependencies
isCritical: false
maxAttempts: 2
inputs:
  InstanceIds:
  - "{{instanceIds}}"
  DocumentName: AWS-RunPowerShellScript
  Parameters:
    commands:
    - msiexec /i {{packageName}}
nextStep: TestPackage
...
```

Gestion des délais d'expiration dans les documents Automation

La propriété `timeoutSeconds` est partagée par toutes les actions Automation. Vous pouvez utiliser cette propriété pour spécifier la valeur de délai d'exécution d'une action. Vous pouvez également modifier la manière dont une action qui arrive à expiration affecte le flux de travail Automation et le statut global de l'exécution. Pour ce faire, définissez également les propriétés partagées `onFailure` et `isCritical` d'une action.

Par exemple, selon votre cas d'utilisation, vous souhaiterez peut-être qu'Automation passe à une autre action sans affecter le statut global d'Automation si une action arrive à expiration. Dans cet exemple, vous spécifiez la durée d'attente avant l'expiration de l'action à l'aide de la propriété `timeoutSeconds`. Vous spécifiez ensuite l'action, ou l'étape, par laquelle Automation doit passer si un délai d'expiration a été spécifié. Spécifiez une valeur en utilisant le format `step:`*`step_name`* de la `onFailure` propriété plutôt que la valeur par défaut de `Abort` . Par défaut, si une action expire, le statut d'exécution d'Automation est `Timed Out` . Pour empêcher un délai d'attente d'affecter le statut d'exécution d'Automation, spécifiez `false` pour la `isCritical` propriété.

L'exemple suivant illustre comment définir les propriétés partagées pour une action décrite dans ce scénario.

YAML

```
- name: verifyImageAvailability
  action: 'aws:waitForAwsResourceProperty'
  timeoutSeconds: 600
  isCritical: false
  onFailure: 'step:getCurrentImageState'
  inputs:
    Service: ec2
    Api: DescribeImages
    ImageIds:
      - '{{ createImage.newImageId }}'
    PropertySelector: '$.Images[0].State'
    DesiredValues:
      - available
  nextStep: copyImage
```

JSON

```
{
    "name": "verifyImageAvailability",
    "action": "aws:waitForAwsResourceProperty",
    "timeoutSeconds": 600,
    "isCritical": false,
    "onFailure": "step:getCurrentImageState",
    "inputs": {
        "Service": "ec2",
        "Api": "DescribeImages",
        "ImageIds": [
            "{{ createImage.newImageId }}"
        ],
        "PropertySelector": "$.Images[0].State",
        "DesiredValues": [
            "available"
        ]
    },
    "nextStep": "copyImage"
}
```

Pour de plus amples informations sur les propriétés partagées par toutes les actions Automation, veuillez consulter Propriétés partagées par toutes les actions (p. 387).

Appel d'autres services AWS à partir d'un flux de travail Systems Manager Automation

Vous pouvez appeler d'autres services AWS et d'autres fonctionnalités Systems Manager dans votre flux de travail Automation à l'aide des actions Automation suivantes dans vos documents Automation.

- aws:executeAwsApi : Cette action Automation appelle et exécute des actions AWS d'API . La plupart des actions d'API sont prises en charge, mais pas toutes les actions d'API ont été testées. Par exemple, les actions d'API suivantes sont prises en charge : CreateImage, supprimer un compartiment RebootDBInstance , et CreateGroups , pour n'en citer que quelques-uns. Les actions API de streaming telles que l'action Get Object ne sont pas prises en charge.
- aws:waitForAwsResourceProperty : Cette action Automation permet à votre flux de travail d'attendre un état de ressource ou d'événement spécifique avant de poursuivre le flux de travail. Par exemple, vous pouvez utiliser cette action avec l'action Amazon Relational Database Service Amazon RDS d'API DescribeDBInstances () pour suspendre un flux de travail Automation afin qu'une instance de base de données ait le temps de démarrer.
- aws:assertAwsResourceProperty : Cette action Automation vous permet d'imposer un état de ressource ou d'événement spécifique pour une étape d'Automation spécifique. Par exemple, vous pouvez spécifier qu'une étape d'automatisation doit attendre le démarrage d'une instance EC2. Elle appellera ensuite l'action Amazon EC2 DescribeInstanceStatus d'API avec la DesiredValue propriété de `running` . Cela garantit que le flux de travail Automation attend une instance en cours d'exécution, puis continue lorsque l'instance est en cours d'exécution.

Voici un exemple de document d'Automation au format YAML qui utilise l'action aws:executeAwsApi pour désactiver les autorisations de lecture et d'écriture dans un compartiment S3.

```
---
description: Disable S3-Bucket's public WriteRead access via private ACL
schemaVersion: "0.3"
assumeRole: "{{ AutomationAssumeRole }}"
parameters:
  S3BucketName:
    type: String
    description: (Required) S3 Bucket subject to access restriction
  AutomationAssumeRole:
    type: String
    description: (Optional) The ARN of the role that allows Automation to perform the
 actions on your behalf.
    default: ""
mainSteps:
- name: DisableS3BucketPublicReadWrite
  action: aws:executeAwsApi
  inputs:
    Service: s3
    Api:  PutBucketAcl
    Bucket: "{{S3BucketName}}"
    ACL: private
  isEnd: true
...
```

Voici un exemple de document d'Automation au format YAML qui utilise les trois actions. Ce document effectue les opérations suivantes :

- Utilise l'action aws:executeAwsApi pour appeler l'action Amazon EC2 DescribeImages d'API afin d'obtenir le nom d'une AMI Windows Server 2016 spécifique. Il génère l'ID d'image en tant que `ImageId`.
- Il utilise l'action aws:executeAwsApi pour appeler l'action Amazon EC2 RunInstances d'API afin de lancer une instance qui utilise l' `ImageId` de l'étape précédente. Il génère l'ID d'instance en tant que `InstanceId`.
- Utilise l'action aws:waitForAwsResourceProperty pour interroger l'action Amazon EC2 DescribeInstanceStatus d'API afin d'attendre que l'instance atteigne l'`running`état. L'action arrive à expiration dans les 60 secondes. L'étape arrive à expiration si l'état de l'instance ne parvient pas à atteindre `running` après 60 secondes d'interrogation.

- Il utilise l'action aws:assertAwsResourceProperty pour appeler l'action Amazon EC2 DescribeInstanceStatus d'API afin de confirmer que l'instance est à l'`running`état . L'étape échoue si l'état de l'instance n'est pas `running`.

```
---
description: Sample Automation Document Using AWS API Actions
schemaVersion: '0.3'
assumeRole: "{{ AutomationAssumeRole }}"
parameters:
  AutomationAssumeRole:
    type: String
    description: "(Optional) The ARN of the role that allows Automation to perform the
 actions on your behalf."
    default: ''
  ImageName:
    type: String
    description: "(Optional) Image Name to launch EC2 instance with."
    default: "Windows_Server-2016-English-Full-Base-2018.07.11"
mainSteps:
- name: getImageId
  action: aws:executeAwsApi
  inputs:
    Service: ec2
    Api: DescribeImages
    Filters:
    - Name: "name"
      Values:
      - "{{ ImageName }}"
  outputs:
  - Name: ImageId
    Selector: "$.Images[0].ImageId"
    Type: "String"
- name: launchOneInstance
  action: aws:executeAwsApi
  inputs:
    Service: ec2
    Api: RunInstances
    ImageId: "{{ getImageId.ImageId }}"
    MaxCount: 1
    MinCount: 1
  outputs:
  - Name: InstanceId
    Selector: "$.Instances[0].InstanceId"
    Type: "String"
- name: waitUntilInstanceStateRunning
  action: aws:waitForAwsResourceProperty
  # timeout is strongly encouraged for action - aws:waitForAwsResourceProperty
  timeoutSeconds: 60
  inputs:
    Service: ec2
    Api: DescribeInstanceStatus
    InstanceIds:
    - "{{ launchOneInstance.InstanceId }}"
    PropertySelector: "$.InstanceStatuses[0].InstanceState.Name"
    DesiredValues:
    - running
- name: assertInstanceStateRunning
  action: aws:assertAwsResourceProperty
  inputs:
    Service: ec2
    Api: DescribeInstanceStatus
    InstanceIds:
    - "{{ launchOneInstance.InstanceId }}"
```

```
    PropertySelector: "$.InstanceStatuses[0].InstanceState.Name"
    DesiredValues:
    - running
outputs:
- "launchOneInstance.InstanceId"
...
```

Utilisation des entrées et des sorties

Chacune des actions Automation décrites précédemment vous permet d'appeler une action d'API spécifique en définissant l'espace de noms de service, l'action d'API, les paramètres d'entrée et les paramètres de sortie. Les entrées sont définies par l'action d'API que vous choisissez. Vous pouvez afficher les actions d'API (également appelées méthodes) en choisissant un service dans le panneau de navigation de gauche sur la page Services Reference (Référence des services) suivante. Choisissez une méthode dans la section Client pour le service que vous voulez appeler. Par exemple, toutes les actions d'API (méthodes) pour Amazon Relational Database Service (Amazon RDS) sont répertoriées sur la page suivante : Méthodes Amazon RDS.

Vous pouvez consulter le schéma de chaque action Automation dans les emplacements suivants :

- aws:assertAwsResourceProperty – Affirmer un état de ressource AWS ou un état d'événement (p. 395)
- aws:executeAwsApi – Appeler et exécuter des actions de l'API AWS (p. 416)
- aws:waitForAwsResourceProperty – Attendre une propriété de ressource AWS (p. 435)

Les schémas incluent les descriptions des champs obligatoires pour l'utilisation de chaque action.

Utilisation des champs Selector/PropertySelector

Chaque action Automation nécessite que vous spécifiiez un `Selector` de sortie (pour aws:executeAwsApi) ou un `PropertySelector` (pour aws:assertAwsResourceProperty et aws:waitForAwsResourceProperty). Ces champs sont utilisés pour traiter la réponse JSON à partir d'une action d'API AWS. Ces champs utilisent la JSONPath syntaxe.

Voici un exemple pour vous aider à illustrer ce concept pour l'action `aws:executeAwsAPi` :

```
---
mainSteps:
- name: getImageId
  action: aws:executeAwsApi
  inputs:
    Service: ec2
    Api: DescribeImages
    Filters:
      - Name: "name"
        Values:
          - "{{ ImageName }}"
  outputs:
    - Name: ImageId
      Selector: "$.Images[0].ImageId"
      Type: "String"
...
```

À l'`aws:executeAwsApi`étape `getImageId` , le flux de travail appelle l'action `DescribeImages` d'API et reçoit une réponse d' `ec2` . Le flux de travail s'applique ensuite `Selector - "$.Images[0].ImageId"` à la réponse de l'API et attribue la valeur sélectionnée à la `ImageId` variable de sortie. D'autres étapes dans le même flux de travail d'automatisation peuvent utiliser la valeur `ImageId` en spécifiant `"{{ getImageId.ImageId }}"`.

Voici un exemple pour vous aider à illustrer ce concept pour l'action `aws:waitForAwsResourceProperty` :

```
---
- name: waitUntilInstanceStateRunning
  action: aws:waitForAwsResourceProperty
  # timeout is strongly encouraged for action - aws:waitForAwsResourceProperty
  timeoutSeconds: 60
  inputs:
    Service: ec2
    Api: DescribeInstanceStatus
    InstanceIds:
    - "{{ launchOneInstance.InstanceId }}"
    PropertySelector: "$.InstanceStatuses[0].InstanceState.Name"
    DesiredValues:
    - running
...
```

À l'`aws:waitForAwsResourceProperty`étape `waitUntilInstanceStateRunning` , le flux de travail appelle l'action `DescribeInstanceStatus` d'API et reçoit une réponse d' `ec2` . Le flux de travail s'applique ensuite `PropertySelector - "$.InstanceStatuses[0].InstanceState.Name"` à la réponse et vérifie si la valeur renvoyée correspond à une valeur de la `DesiredValues` liste (dans ce cas`running`). L'étape répète le processus jusqu'à ce que la réponse renvoie l'état d'instance `running`.

Utilisation d' JSONPath dans un flux de travail Automation

Une JSONPath expression est une chaîne commençant par « $ ». qui est utilisée pour sélectionner un ou plusieurs composants au sein d'un élément JSON. La liste suivante inclut des informations sur JSONPath les opérateurs pris en charge par Systems Manager Automation :

- Enfant à notation point (.) : A utiliser avec un objet JSON. Cet opérateur sélectionne la valeur d'une clé spécifique.
- Analyse approfondie (..) : A utiliser avec un élément JSON. Cet opérateur numérise l'élément JSON niveau par niveau et sélectionne une liste de valeurs avec la clé spécifique. Notez que le type de retour de cet opérateur est toujours un tableau JSON. Dans le contexte d'un type de sortie d'étape d'automatisation, l'opérateur peut être StringList ou MapList .
- Index de tableau ([]) : A utiliser avec un tableau JSON. Cet opérateur obtient la valeur d'un index spécifique.

Afin de mieux comprendre JSONPath les opérateurs, vérifiez la réponse JSON suivante à partir de l'action `DescribeInstances` d'API ec2. Sous cette réponse figurent plusieurs exemples qui illustrent différents résultats en appliquant différentes JSONPath expressions à la réponse à partir de l'action `DescribeInstances` d'API.

```
{
    "NextToken": "abcdefg",
    "Reservations": [
        {
            "OwnerId": "123456789012",
            "ReservationId": "r-abcd12345678910",
            "Instances": [
                {
                    "ImageId": "ami-12345678",
                    "BlockDeviceMappings": [
                        {
                            "Ebs": {
                                "DeleteOnTermination": true,
                                "Status": "attached",
                                "VolumeId": "vol-000000000000"
                            },
                            "DeviceName": "/dev/xvda"
                        }
                    ],
```

```
                    "State": {
                        "Code": 16,
                        "Name": "running"
                    }
                }
            ],
            "Groups": []
        },
        {
            "OwnerId": "123456789012",
            "ReservationId": "r-12345678910abcd",
            "Instances": [
                {
                    "ImageId": "ami-12345678",
                    "BlockDeviceMappings": [
                        {
                            "Ebs": {
                                "DeleteOnTermination": true,
                                "Status": "attached",
                                "VolumeId": "vol-111111111111"
                            },
                            "DeviceName": "/dev/xvda"
                        }
                    ],
                    "State": {
                        "Code": 80,
                        "Name": "stopped"
                    }
                }
            ],
            "Groups": []
        }
    ]
}
```

JSONPathExemple 1 : Obtenir une chaîne spécifique à partir d'une réponse JSON

```
JSONPath:
$.Reservations[0].Instances[0].ImageId

Returns:
"ami-12345678"

Type: String
```

JSONPathExemple 2 : Obtention d'une valeur booléenne spécifique à partir d'une réponse JSON

```
JSONPath:
$.Reservations[0].Instances[0].BlockDeviceMappings[0].Ebs.DeleteOnTermination

Returns:
true

Type: Boolean
```

JSONPathExemple 3 : Obtention d'un entier spécifique à partir d'une réponse JSON

```
JSONPath:
$.Reservations[0].Instances[0].State.Code

Returns:
16
```

```
Type: Integer
```

JSONPathExemple 4 : Analyse approfondie d'une réponse JSON, puis obtention de toutes les valeurs pour VolumeId en tant queStringList

```
JSONPath:
$.Reservations..BlockDeviceMappings..VolumeId

Returns:
[
   "vol-000000000000",
   "vol-111111111111"
]

Type: StringList
```

JSONPathExemple 5 : Obtention d'un BlockDeviceMappings objet spécifique en tant queStringMap

```
JSONPath:
$.Reservations[0].Instances[0].BlockDeviceMappings[0]

Returns:
{
   "Ebs" : {
      "DeleteOnTermination" : true,
      "Status" : "attached",
      "VolumeId" : "vol-000000000000"
   },
   "DeviceName" : "/dev/xvda"
}

Type: StringMap
```

JSONPathExemple 6 : Analyse approfondie d'une réponse JSON, puis obtention de tous les objets d'état sous la forme d'uneMapList

```
JSONPath:
$.Reservations..Instances..State

Returns:
[
   {
      "Code" : 16,
      "Name" : "running"
   },
   {
      "Code" : 80,
      "Name" : "stopped"
   }
]

Type: MapList
```

> **Important**
>
> Si vous exécutez un flux de travail d’automatisation qui appelle d'autres services à l'aide d'un rôle de service AWS Identity and Access Management (IAM), le rôle de service doit être configuré avec l'autorisation d'appeler ces services. Cette exigence s'applique à tous les documents AWS Automation (documents `AWS-*`) comme les documents `AWS-ConfigureS3BucketLogging`, `AWS-CreateDynamoDBBackup` et `AWS-RestartEC2Instance`, pour n'en citer que quelques-uns. Cette exigence s'applique également à tous les documents Automation personnalisés

que vous créez qui appellent d'autres services AWS à l'aide d'actions qui appellent d'autres services. Par exemple, si vous utilisez les actions `aws:executeAwsApi`, `aws:createStack` ou `aws:copyImage`, vous devez configurer le rôle de service avec l'autorisation d'appeler ces services. Vous pouvez activer des autorisations à d'autres services AWS en ajoutant une stratégie IAM en ligne au rôle. Pour de plus amples informations, veuillez consulter (Facultatif) Ajoutez une stratégie en ligne Automation pour appeler d'autres services AWS (p. 311).

Exemple de procédure : Démarrer une Amazon RDS instance à partir d'un flux de travail Systems Manager Automation

Cet exemple de procédure vous montre comment créer et exécuter un document Automation au format YAML qui utilise les trois actions d'API pour voir si une instance de base de données Amazon Relational Database Service (Amazon RDS) est en cours d'exécution. Si l'instance n'est pas en cours d'exécution, le flux de travail la démarre.

Pour appeler une action d'API Amazon RDS à partir de Systems Manager Automation

1. Ouvrez le fichier dans un éditeur de texte et collez le contenu suivant du document d'Automation dans le fichier. Spécifiez un rôle d'automatisation et l'ID d'instance à vérifier. Vous ajouterez les mainSteps actions ultérieurement.

```
---
description: Start RDS instance
schemaVersion: "0.3"
assumeRole: "The_Automation_role_to_use_when_running_the_document"
parameters:
  InstanceId: The_instance_ID_to_start
    type: String
    description: (Required) RDS instance ID to start
  AutomationAssumeRole:
    type: String
    description: (Optional) The ARN of the role that allows Automation to perform the
 actions on your behalf.
    default: ""
mainSteps:
```

2. Pour la première étape du flux de travail, vous devez déterminer si l'instance est déjà en cours d'exécution. Vous pouvez utiliser l'action aws:assertAwsResourceProperty afin de déterminer et d'imposer un état d'instance spécifique. Avant de pouvoir ajouter l'action aws:assertAwsResourceProperty au document, vous devez déterminer et spécifier les entrées obligatoires. La liste suivante décrit comment déterminer et spécifier les entrées requises. Vous pouvez consulter un exemple de la marche à suivre pour indiquer ces informations dans le document d'Automation suivant la liste.

   a. Affichez le schéma pour voir toutes les entrées disponibles pour l'action aws:assertAwsResourceProperty – Affirmer un état de ressource AWS ou un état d'événement (p. 395).
   b. Déterminez l'espace de noms du service à appeler. Vous pouvez afficher la liste des espaces de noms de service AWS dans les ARN (Amazon Resource Name) et espaces de noms du service AWS dans Référence générale d'Amazon Web Services. L'espace de noms pour Amazon RDS est `rds`.
   c. Déterminez quelle action d'API Amazon RDS vous permet d'afficher le statut d'une instance de base de données. Vous pouvez afficher les actions d'API (également appelées méthodes) sur la page Méthodes Amazon RDS.
   d. Spécifiez un ou plusieurs paramètres de demande pour l'action DescribeDBInstances d'API. Par exemple, cette action utilise le paramètre de demande `DBInstanceIdentifier`.
   e. Déterminez une ou plusieurs PropertySelectors . Un PropertySelector est un objet de réponse qui est renvoyé par la demande de cette action d'API. Par exemple, sur la page Méthodes Amazon

RDS. Choisissez la méthode describe_db_instances et faites défiler la page jusqu'à la section Response Structure (Structure d'une réponse). DBInstancesest répertorié en tant qu'objet de réponse. Dans le cadre de cette procédure, spécifiez `DBInstances` et `DBInstanceStatus` comme PropertySelectors . N'oubliez pas que PropertySelectors les données sont entrées à l'aide d' JSONPath . Cela signifie que vous formatez les informations dans le document Automation comme suit :

```
PropertySelector: "$.DBInstances[0].DBInstanceStatus".
```

f. Spécifiez une ou plusieurs DesiredValues . Si vous ne connaissez pas les valeurs que vous souhaitez spécifier, exécutez l'action DescribeDBInstances d'API pour déterminer les valeurs possibles. Pour cette procédure, spécifiez available et starting.

g. Saisissez les informations que vous avez recueillies dans le document d'Automation, comme indiqué dans l'exemple suivant.

```
---
description: Start RDS instance
schemaVersion: "0.3"
assumeRole: "The_Automation_role_to_use_when_running_the_document"
parameters:
  InstanceId: The_instance_ID_to_start
    type: String
    description: (Required) RDS Instance Id to stop
  AutomationAssumeRole:
    type: String
    description: (Optional) The ARN of the role that allows Automation to perform the
 actions on your behalf.
    default: ""
mainSteps:
  -
    name: AssertNotStartingOrAvailable
    action: aws:assertAwsResourceProperty
    isCritical: false
    onFailure: step:StartInstance
    nextStep: CheckStart
    inputs:
      Service: rds
      Api: DescribeDBInstances
      DBInstanceIdentifier: "{{InstanceId}}"
      PropertySelector: "$.DBInstances[0].DBInstanceStatus"
      DesiredValues: ["available", "starting"]
```

3. Spécifiez une action aws:executeAwsApi dans la mainSteps section pour démarrer l'instance si l'action précédente a déterminé qu'elle n'a pas été lancée.

   a. Affichez le schéma pour voir toutes les entrées disponibles pour aws:executeAwsApi – Appeler et exécuter des actions de l'API AWS (p. 416).

   b. Spécifiez l'action Amazon RDS StartDBInstance d'API pour démarrer l'instance.

   c. Saisissez les informations que vous avez recueillies dans le document d'Automation, comme indiqué dans l'exemple suivant.

```
---
description: Start RDS instance
schemaVersion: "0.3"
assumeRole: "{{ The_Automation_role_to_use_when_running_the_document }}"
parameters:
  InstanceId:
    type: String
    description: (Required) RDS Instance Id to stop
```

```
  AutomationAssumeRole:
    type: String
    description: (Optional) The ARN of the role that allows Automation to perform the
 actions on your behalf.
    default: ""
mainSteps:
  -
    name: AssertNotStartingOrAvailable
    action: aws:assertAwsResourceProperty
    isCritical: false
    onFailure: step:StartInstance
    nextStep: CheckStart
    inputs:
      Service: rds
      Api: DescribeDBInstances
      DBInstanceIdentifier: "{{InstanceId}}"
      PropertySelector: "$.DBInstances[0].DBInstanceStatus"
      DesiredValues: ["available", "starting"]
  -
    name: StartInstance
    action: aws:executeAwsApi
    inputs:
      Service: rds
      Api: StartDBInstance
      DBInstanceIdentifier: "{{InstanceId}}"
```

4. Spécifiez une action aws:waitForAwsResourceProperty dans la mainSteps section pour attendre que l'instance démarre avant de terminer le flux de travail Automation.

   a. Affichez le schéma pour voir toutes les entrées disponibles pour aws:waitForAwsResourceProperty – Attendre une propriété de ressource AWS (p. 435).

   b. Spécifiez l'action Amazon RDS DescribeDBInstances d'API pour déterminer le statut de l'instance.

   c. Spécifiez `$.DBInstances[0].DBInstanceStatus` en tant que `PropertySelector`.

   d. Spécifiez available en tant que `DesiredValue`.

   e. Saisissez les informations que vous avez recueillies dans le document d'Automation, comme indiqué dans l'exemple suivant.

```
---
description: Start RDS instance
schemaVersion: "0.3"
assumeRole: "{{ The_Automation_role_to_use_when_running_the_document }}"
parameters:
  InstanceId:
    type: String
    description: (Required) RDS Instance Id to stop
  AutomationAssumeRole:
    type: String
    description: (Optional) The ARN of the role that allows Automation to perform the
 actions on your behalf.
    default: ""
mainSteps:
  -
    name: AssertNotStartingOrAvailable
    action: aws:assertAwsResourceProperty
    isCritical: false
    onFailure: step:StartInstance
    nextStep: CheckStart
    inputs:
      Service: rds
      Api: DescribeDBInstances
```

```
      DBInstanceIdentifier: "{{InstanceId}}"
      PropertySelector: "$.DBInstances[0].DBInstanceStatus"
      DesiredValues: ["available", "starting"]
  -
    name: StartInstance
    action: aws:executeAwsApi
    inputs:
      Service: rds
      Api: StartDBInstance
      DBInstanceIdentifier: "{{InstanceId}}"
  -
    name: CheckStart
    action: aws:waitForAwsResourceProperty
    onFailure: Abort
    maxAttempts: 10
    timeoutSeconds: 600
    inputs:
      Service: rds
      Api: DescribeDBInstances
      DBInstanceIdentifier: "{{InstanceId}}"
      PropertySelector: "$.DBInstances[0].DBInstanceStatus"
      DesiredValues: ["available"]
    isEnd: true
...
```

5. Enregistrez le fichier sous le nom sample.yaml.
6. Exécutez la commande suivante dans l'AWS CLI pour ajouter le document à votre compte AWS.

```
aws ssm create-document --name sampleDoc --document-type Automation --document-format
 YAML --content file://sample.yaml
```

7. Exécutez la commande suivante pour lancer l'exécution d'Automation avec le document que vous venez de créer. Prenez note de l'ID d'exécution renvoyée par Systems Manager après le démarrage de l'exécution.

```
aws ssm start-automation-execution --document-name sampleDoc
```

8. Exécutez la commande suivante pour afficher l'état de l'exécution.

```
aws ssm get-automation-execution --automation-execution-id automation_execution_id
```

Documents Automation prédéfinis qui appellentAWSAPIs

Systems ManagerL'automatisation inclut les documents SSM Automation prédéfinis suivants qui appellent AWS APIs .

Nom du document	Objectif
AWS-StartRdsInstance	Démarrez une instance Amazon RDS.
AWS-StopRdsInstance	Arrêtez une instance Amazon RDS.
AWS-RebootRdsInstance	Redémarrez une instance Amazon RDS.
AWS-CreateSnapshot	Créez un instantané d'un volume Amazon Elastic Block Store (Amazon EBS).
AWS-DeleteSnapshot	Supprimez un instantané d'un volume Amazon EBS.

Nom du document	Objectif
AWS-ConfigureS3BucketLogging	Activez la journalisation au niveau d'un compartiment Amazon Simple Storage Service (Amazon S3).
AWS-DisableS3BucketPublicReadWrite	Désactivez les autorisations de lecture et d'écriture sur un compartiment S3 à l'aide d'une liste de contrôle d'accès (ACL) privée.
AWS-ConfigureS3BucketVersioning	Activez ou interrompez la gestion des versions dans un compartiment S3.
AWS-DeleteDynamoDbBackup	Supprimez une sauvegarde de table Amazon DynamoDB (DynamoDB).

Cliquez sur les liens dans le tableau ci-dessus ou utilisez la procédure suivante pour afficher plus d'informations sur ces documents Automation dans la console Systems Manager.

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Documents dans le panneau de navigation.
3. Choisissez un document, puis choisissez View details (Afficher les détails).
4. Choisissez l'onglet Content (Contenu).

Exemples de scénarios et solutions de documents Automation personnalisées

Les exemples de documents Automation suivants montrent comment vous pouvez utiliser les actions AWS Systems Manager Automation pour automatiser les tâches courantes de déploiement, de dépannage et de maintenance.

> **Note**
>
> Les exemples de document Automation de cette section sont fournis pour démontrer comment créer des documents Automation personnalisés pour répondre à vos besoins opérationnels spécifiques. Ces documents ne sont pas destinés à être utilisés dans les environnements de production tels qu'ils sont. Cependant, vous pouvez les personnaliser pour votre propre usage.

Échantillons

- Déployer l'architecture VPC et les contrôleurs de domaine Microsoft Active Directory (p. 481)
- Restaurer un volume racine à partir du dernier instantané (p. 500)
- Créer une AMI et une copie inter-régions (p. 508)

Déployer l'architecture VPC et les contrôleurs de domaine Microsoft Active Directory

Pour accroître l'efficacité et normaliser les tâches courantes, vous pouvez choisir d'automatiser les déploiements. C'est particulièrement utile si vous déployez régulièrement la même architecture sur

plusieurs comptes et dans plusieurs régions. L'automatisation des déploiements d'architecture peut également réduire le risque d'erreurs humaines qui peuvent se produire lors du déploiement manuel de l'architecture.AWS Systems Manager Les actions Automation peuvent vous aider à y parvenir.

L'exemple de document AWS Systems Manager Automation suivant effectue ces actions.

- Récupère la dernière image Amazon Machine Image (AMI) Windows Server 2012R2 avec Systems Manager Parameter Store à utiliser lors du lancement des instances EC2 qui seront configurées en tant que contrôleurs de domaine.
- Utilise l'action `aws:executeAwsApi` Automation pour appeler plusieurs actions d'API AWS pour créer l'architecture VPC. Les instances de contrôleur de domaine sont lancées dans des sous-réseaux privés et se connectent à Internet à l'aide d'une passerelle NAT. Cela permet à l' Agent SSM sur les instances d'accéder aux points de terminaison Systems Manager requis.
- Il utilise l'action Automation `aws:waitForAwsResourceProperty` pour confirmer que les instances lancées par l'action précédente sont `Online` pour AWS Systems Manager.
- Il utilise l'action Automation `aws:runCommand` pour configurer les instances lancées en tant que contrôleurs de domaine Microsoft Active Directory.

YAML

```
---
description: Custom Automation Deployment Sample
schemaVersion: '0.3'
parameters:
  AutomationAssumeRole:
    type: String
    default: ''
    description: >-
      (Optional) The ARN of the role that allows Automation to perform the
      actions on your behalf. If no role is specified, Systems Manager
      Automation uses your IAM permissions to run this document.
mainSteps:
  - name: getLatestWindowsAmi
    action: aws:executeAwsApi
    onFailure: Abort
    inputs:
      Service: ssm
      Api: GetParameter
      Name: >-
        /aws/service/ami-windows-latest/Windows_Server-2012-R2_RTM-English-64Bit-Base
    outputs:
      - Name: amiId
        Selector: $.Parameter.Value
        Type: String
    nextStep: createSSMInstanceRole
  - name: createSSMInstanceRole
    action: aws:executeAwsApi
    onFailure: Abort
    inputs:
      Service: iam
      Api: CreateRole
      AssumeRolePolicyDocument: >-
        {"Version":"2012-10-17","Statement":[{"Effect":"Allow","Principal":{"Service":
["ec2.amazonaws.com"]},"Action":["sts:AssumeRole"]}]}
      RoleName: sampleSSMInstanceRole
    nextStep: attachManagedSSMPolicy
  - name: attachManagedSSMPolicy
    action: aws:executeAwsApi
    onFailure: Abort
    inputs:
```

```
    Service: iam
    Api: AttachRolePolicy
    PolicyArn: 'arn:aws:iam::aws:policy/service-role/AmazonSSMManagedInstanceCore'
    RoleName: sampleSSMInstanceRole
  nextStep: createSSMInstanceProfile
- name: createSSMInstanceProfile
  action: aws:executeAwsApi
  onFailure: Abort
  inputs:
    Service: iam
    Api: CreateInstanceProfile
    InstanceProfileName: sampleSSMInstanceRole
  outputs:
    - Name: instanceProfileArn
      Selector: $.InstanceProfile.Arn
      Type: String
  nextStep: addSSMInstanceRoleToProfile
- name: addSSMInstanceRoleToProfile
  action: aws:executeAwsApi
  onFailure: Abort
  inputs:
    Service: iam
    Api: AddRoleToInstanceProfile
    InstanceProfileName: sampleSSMInstanceRole
    RoleName: sampleSSMInstanceRole
  nextStep: createVpc
- name: createVpc
  action: aws:executeAwsApi
  onFailure: Abort
  inputs:
    Service: ec2
    Api: CreateVpc
    CidrBlock: 10.0.100.0/22
  outputs:
    - Name: vpcId
      Selector: $.Vpc.VpcId
      Type: String
  nextStep: getMainRtb
- name: getMainRtb
  action: aws:executeAwsApi
  onFailure: Abort
  inputs:
    Service: ec2
    Api: DescribeRouteTables
    Filters:
      - Name: vpc-id
        Values:
          - '{{ createVpc.vpcId }}'
  outputs:
    - Name: mainRtbId
      Selector: '$.RouteTables[0].RouteTableId'
      Type: String
  nextStep: verifyMainRtb
- name: verifyMainRtb
  action: aws:assertAwsResourceProperty
  onFailure: Abort
  inputs:
    Service: ec2
    Api: DescribeRouteTables
    RouteTableIds:
      - '{{ getMainRtb.mainRtbId }}'
    PropertySelector: '$.RouteTables[0].Associations[0].Main'
    DesiredValues:
      - 'True'
  nextStep: createPubSubnet
- name: createPubSubnet
```

```
    action: aws:executeAwsApi
    onFailure: Abort
    inputs:
      Service: ec2
      Api: CreateSubnet
      CidrBlock: 10.0.103.0/24
      AvailabilityZone: us-west-2c
      VpcId: '{{ createVpc.vpcId }}'
    outputs:
      - Name: pubSubnetId
        Selector: $.Subnet.SubnetId
        Type: String
    nextStep: createPubRtb
  - name: createPubRtb
    action: aws:executeAwsApi
    onFailure: Abort
    inputs:
      Service: ec2
      Api: CreateRouteTable
      VpcId: '{{ createVpc.vpcId }}'
    outputs:
      - Name: pubRtbId
        Selector: $.RouteTable.RouteTableId
        Type: String
    nextStep: createIgw
  - name: createIgw
    action: aws:executeAwsApi
    onFailure: Abort
    inputs:
      Service: ec2
      Api: CreateInternetGateway
    outputs:
      - Name: igwId
        Selector: $.InternetGateway.InternetGatewayId
        Type: String
    nextStep: attachIgw
  - name: attachIgw
    action: aws:executeAwsApi
    onFailure: Abort
    inputs:
      Service: ec2
      Api: AttachInternetGateway
      InternetGatewayId: '{{ createIgw.igwId }}'
      VpcId: '{{ createVpc.vpcId }}'
    nextStep: allocateEip
  - name: allocateEip
    action: aws:executeAwsApi
    onFailure: Abort
    inputs:
      Service: ec2
      Api: AllocateAddress
      Domain: vpc
    outputs:
      - Name: eipAllocationId
        Selector: $.AllocationId
        Type: String
    nextStep: createNatGw
  - name: createNatGw
    action: aws:executeAwsApi
    onFailure: Abort
    inputs:
      Service: ec2
      Api: CreateNatGateway
      AllocationId: '{{ allocateEip.eipAllocationId }}'
      SubnetId: '{{ createPubSubnet.pubSubnetId }}'
    outputs:
```

```
    - Name: natGwId
      Selector: $.NatGateway.NatGatewayId
      Type: String
  nextStep: verifyNatGwAvailable
- name: verifyNatGwAvailable
  action: aws:waitForAwsResourceProperty
  timeoutSeconds: 150
  inputs:
    Service: ec2
    Api: DescribeNatGateways
    NatGatewayIds:
      - '{{ createNatGw.natGwId }}'
    PropertySelector: '$.NatGateways[0].State'
    DesiredValues:
      - available
  nextStep: createNatRoute
- name: createNatRoute
  action: aws:executeAwsApi
  onFailure: Abort
  inputs:
    Service: ec2
    Api: CreateRoute
    DestinationCidrBlock: 0.0.0.0/0
    NatGatewayId: '{{ createNatGw.natGwId }}'
    RouteTableId: '{{ getMainRtb.mainRtbId }}'
  nextStep: createPubRoute
- name: createPubRoute
  action: aws:executeAwsApi
  onFailure: Abort
  inputs:
    Service: ec2
    Api: CreateRoute
    DestinationCidrBlock: 0.0.0.0/0
    GatewayId: '{{ createIgw.igwId }}'
    RouteTableId: '{{ createPubRtb.pubRtbId }}'
  nextStep: setPubSubAssoc
- name: setPubSubAssoc
  action: aws:executeAwsApi
  onFailure: Abort
  inputs:
    Service: ec2
    Api: AssociateRouteTable
    RouteTableId: '{{ createPubRtb.pubRtbId }}'
    SubnetId: '{{ createPubSubnet.pubSubnetId }}'
- name: createDhcpOptions
  action: aws:executeAwsApi
  onFailure: Abort
  inputs:
    Service: ec2
    Api: CreateDhcpOptions
    DhcpConfigurations:
      - Key: domain-name-servers
        Values:
          - '10.0.100.50,10.0.101.50'
      - Key: domain-name
        Values:
          - sample.com
  outputs:
    - Name: dhcpOptionsId
      Selector: $.DhcpOptions.DhcpOptionsId
      Type: String
  nextStep: createDCSubnet1
- name: createDCSubnet1
  action: aws:executeAwsApi
  onFailure: Abort
  inputs:
```

```
    Service: ec2
    Api: CreateSubnet
    CidrBlock: 10.0.100.0/24
    AvailabilityZone: us-west-2a
    VpcId: '{{ createVpc.vpcId }}'
  outputs:
    - Name: firstSubnetId
      Selector: $.Subnet.SubnetId
      Type: String
  nextStep: createDCSubnet2
- name: createDCSubnet2
  action: aws:executeAwsApi
  onFailure: Abort
  inputs:
    Service: ec2
    Api: CreateSubnet
    CidrBlock: 10.0.101.0/24
    AvailabilityZone: us-west-2b
    VpcId: '{{ createVpc.vpcId }}'
  outputs:
    - Name: secondSubnetId
      Selector: $.Subnet.SubnetId
      Type: String
  nextStep: createDCSecGroup
- name: createDCSecGroup
  action: aws:executeAwsApi
  onFailure: Abort
  inputs:
    Service: ec2
    Api: CreateSecurityGroup
    GroupName: SampleDCSecGroup
    Description: Security Group for Sample Domain Controllers
    VpcId: '{{ createVpc.vpcId }}'
  outputs:
    - Name: dcSecGroupId
      Selector: $.GroupId
      Type: String
  nextStep: authIngressDCTraffic
- name: authIngressDCTraffic
  action: aws:executeAwsApi
  onFailure: Abort
  inputs:
    Service: ec2
    Api: AuthorizeSecurityGroupIngress
    GroupId: '{{ createDCSecGroup.dcSecGroupId }}'
    IpPermissions:
      - FromPort: -1
        IpProtocol: '-1'
        IpRanges:
          - CidrIp: 0.0.0.0/0
            Description: Allow all traffic between Domain Controllers
  nextStep: verifyInstanceProfile
- name: verifyInstanceProfile
  action: aws:waitForAwsResourceProperty
  maxAttempts: 5
  onFailure: Abort
  inputs:
    Service: iam
    Api: ListInstanceProfilesForRole
    RoleName: sampleSSMInstanceRole
    PropertySelector: '$.InstanceProfiles[0].Arn'
    DesiredValues:
      - '{{ createSSMInstanceProfile.instanceProfileArn }}'
  nextStep: iamEventualConsistency
- name: iamEventualConsistency
  action: aws:sleep
```

```
    inputs:
      Duration: PT2M
    nextStep: launchDC1
  - name: launchDC1
    action: aws:executeAwsApi
    onFailure: Abort
    inputs:
      Service: ec2
      Api: RunInstances
      BlockDeviceMappings:
        - DeviceName: /dev/sda1
          Ebs:
            DeleteOnTermination: true
            VolumeSize: 50
            VolumeType: gp2
        - DeviceName: xvdf
          Ebs:
            DeleteOnTermination: true
            VolumeSize: 100
            VolumeType: gp2
      IamInstanceProfile:
        Arn: '{{ createSSMInstanceProfile.instanceProfileArn }}'
      ImageId: '{{ getLatestWindowsAmi.amiId }}'
      InstanceType: t2.micro
      MaxCount: 1
      MinCount: 1
      PrivateIpAddress: 10.0.100.50
      SecurityGroupIds:
        - '{{ createDCSecGroup.dcSecGroupId }}'
      SubnetId: '{{ createDCSubnet1.firstSubnetId }}'
      TagSpecifications:
        - ResourceType: instance
          Tags:
            - Key: Name
              Value: SampleDC1
    outputs:
      - Name: pdcInstanceId
        Selector: '$.Instances[0].InstanceId'
        Type: String
    nextStep: launchDC2
  - name: launchDC2
    action: aws:executeAwsApi
    onFailure: Abort
    inputs:
      Service: ec2
      Api: RunInstances
      BlockDeviceMappings:
        - DeviceName: /dev/sda1
          Ebs:
            DeleteOnTermination: true
            VolumeSize: 50
            VolumeType: gp2
        - DeviceName: xvdf
          Ebs:
            DeleteOnTermination: true
            VolumeSize: 100
            VolumeType: gp2
      IamInstanceProfile:
        Arn: '{{ createSSMInstanceProfile.instanceProfileArn }}'
      ImageId: '{{ getLatestWindowsAmi.amiId }}'
      InstanceType: t2.micro
      MaxCount: 1
      MinCount: 1
      PrivateIpAddress: 10.0.101.50
      SecurityGroupIds:
        - '{{ createDCSecGroup.dcSecGroupId }}'
```

```
    SubnetId: '{{ createDCSubnet2.secondSubnetId }}'
    TagSpecifications:
      - ResourceType: instance
        Tags:
          - Key: Name
            Value: SampleDC2
  outputs:
    - Name: adcInstanceId
      Selector: '$.Instances[0].InstanceId'
      Type: String
  nextStep: verifyDCInstanceState
- name: verifyDCInstanceState
  action: aws:waitForAwsResourceProperty
  inputs:
    Service: ec2
    Api: DescribeInstanceStatus
    IncludeAllInstances: true
    InstanceIds:
      - '{{ launchDC1.pdcInstanceId }}'
      - '{{ launchDC2.adcInstanceId }}'
    PropertySelector: '$.InstanceStatuses[0].InstanceState.Name'
    DesiredValues:
      - running
  nextStep: verifyInstancesOnlineSSM
- name: verifyInstancesOnlineSSM
  action: aws:waitForAwsResourceProperty
  timeoutSeconds: 600
  inputs:
    Service: ssm
    Api: DescribeInstanceInformation
    InstanceInformationFilterList:
      - key: InstanceIds
        valueSet:
          - '{{ launchDC1.pdcInstanceId }}'
          - '{{ launchDC2.adcInstanceId }}'
    PropertySelector: '$.InstanceInformationList[0].PingStatus'
    DesiredValues:
      - Online
  nextStep: installADRoles
- name: installADRoles
  action: aws:runCommand
  inputs:
    DocumentName: AWS-RunPowerShellScript
    InstanceIds:
      - '{{ launchDC1.pdcInstanceId }}'
      - '{{ launchDC2.adcInstanceId }}'
    Parameters:
      commands: |-
        try {
            Install-WindowsFeature -Name AD-Domain-Services -IncludeManagementTools
        }
        catch {
            Write-Error "Failed to install ADDS Role."
        }
  nextStep: setAdminPassword
- name: setAdminPassword
  action: aws:runCommand
  inputs:
    DocumentName: AWS-RunPowerShellScript
    InstanceIds:
      - '{{ launchDC1.pdcInstanceId }}'
    Parameters:
      commands:
        - net user Administrator "sampleAdminPass123!"
  nextStep: createForest
- name: createForest
```

```
    action: aws:runCommand
    inputs:
      DocumentName: AWS-RunPowerShellScript
      InstanceIds:
        - '{{ launchDC1.pdcInstanceId }}'
      Parameters:
        commands: |-
          $dsrmPass = 'sample123!' | ConvertTo-SecureString -asPlainText -Force
          try {
              Install-ADDSForest -DomainName "sample.com" -DomainMode 6 -
ForestMode 6 -InstallDNS -DatabasePath "D:\NTDS" -SysvolPath "D:\SYSVOL" -
SafeModeAdministratorPassword $dsrmPass -Force
          }
          catch {
              Write-Error $_
          }
          try {
              Add-DnsServerForwarder -IPAddress "10.0.100.2"
          }
          catch {
              Write-Error $_
          }
    nextStep: associateDhcpOptions
  - name: associateDhcpOptions
    action: aws:executeAwsApi
    onFailure: Abort
    inputs:
      Service: ec2
      Api: AssociateDhcpOptions
      DhcpOptionsId: '{{ createDhcpOptions.dhcpOptionsId }}'
      VpcId: '{{ createVpc.vpcId }}'
    nextStep: waitForADServices
  - name: waitForADServices
    action: aws:sleep
    inputs:
      Duration: PT1M
    nextStep: promoteADC
  - name: promoteADC
    action: aws:runCommand
    inputs:
      DocumentName: AWS-RunPowerShellScript
      InstanceIds:
        - '{{ launchDC2.adcInstanceId }}'
      Parameters:
        commands: |-
          ipconfig /renew
          $dsrmPass = 'sample123!' | ConvertTo-SecureString -asPlainText -Force
          $domAdminUser = "sample\Administrator"
          $domAdminPass = "sampleAdminPass123!" | ConvertTo-SecureString -asPlainText -
Force
          $domAdminCred = New-Object
 System.Management.Automation.PSCredential($domAdminUser,$domAdminPass)

          try {
              Install-ADDSDomainController -DomainName "sample.com" -InstallDNS -
DatabasePath "D:\NTDS" -SysvolPath "D:\SYSVOL" -SafeModeAdministratorPassword $dsrmPass
 -Credential $domAdminCred -Force
          }
          catch {
              Write-Error $_
          }
```

JSON

```
{
  "description": "Custom Automation Deployment Sample",
  "schemaVersion": "0.3",
  "assumeRole": "{{ AutomationAssumeRole }}",
  "parameters": {
    "AutomationAssumeRole": {
      "type": "String",
      "description": "(Optional) The ARN of the role that allows Automation to perform
 the actions on your behalf. If no role is specified, Systems Manager Automation uses
 your IAM permissions to execute this document.",
      "default": ""
    }
  },
  "mainSteps": [
    {
      "name": "getLatestWindowsAmi",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ssm",
        "Api": "GetParameter",
        "Name": "/aws/service/ami-windows-latest/Windows_Server-2012-R2_RTM-
English-64Bit-Base"
      },
      "outputs": [
        {
          "Name": "amiId",
          "Selector": "$.Parameter.Value",
          "Type": "String"
        }
      ],
      "nextStep": "createSSMInstanceRole"
    },
    {
      "name": "createSSMInstanceRole",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "iam",
        "Api": "CreateRole",
        "AssumeRolePolicyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":
[{\"Effect\":\"Allow\",\"Principal\":{\"Service\":[\"ec2.amazonaws.com\"]},\"Action\":
[\"sts:AssumeRole\"]}]}",
        "RoleName": "sampleSSMInstanceRole"
      },
      "nextStep": "attachManagedSSMPolicy"
    },
    {
      "name": "attachManagedSSMPolicy",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "iam",
        "Api": "AttachRolePolicy",
        "PolicyArn": "arn:aws:iam::aws:policy/service-role/
AmazonSSMManagedInstanceCore",
        "RoleName": "sampleSSMInstanceRole"
      },
      "nextStep": "createSSMInstanceProfile"
    },
    {
      "name": "createSSMInstanceProfile",
      "action":"aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "iam",
```

```
        "Api": "CreateInstanceProfile",
        "InstanceProfileName": "sampleSSMInstanceRole"
      },
      "outputs": [
        {
          "Name": "instanceProfileArn",
          "Selector": "$.InstanceProfile.Arn",
          "Type": "String"
        }
      ],
      "nextStep": "addSSMInstanceRoleToProfile"
    },
    {
      "name": "addSSMInstanceRoleToProfile",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "iam",
        "Api": "AddRoleToInstanceProfile",
        "InstanceProfileName": "sampleSSMInstanceRole",
        "RoleName": "sampleSSMInstanceRole"
      },
      "nextStep": "createVpc"
    },
    {
      "name": "createVpc",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "CreateVpc",
        "CidrBlock": "10.0.100.0/22"
      },
      "outputs": [
        {
          "Name": "vpcId",
          "Selector": "$.Vpc.VpcId",
          "Type": "String"
        }

      "nextStep": "getMainRtb"
    },
    {
      "name": "getMainRtb",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "DescribeRouteTables",
        "Filters": [
          {
            "Name": "vpc-id",
            "Values": ["{{ createVpc.vpcId }}"]
          }
        ]
      },
      "outputs": [
        {
          "Name": "mainRtbId",
          "Selector": "$.RouteTables[0].RouteTableId",
          "Type": "String"
        }
      ],
      "nextStep": "verifyMainRtb"
    },
    {
```

```
      "name": "verifyMainRtb",
      "action": "aws:assertAwsResourceProperty",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "DescribeRouteTables",
        "RouteTableIds": ["{{ getMainRtb.mainRtbId }}"],
        "PropertySelector": "$.RouteTables[0].Associations[0].Main",
        "DesiredValues": ["True"]
      },
      "nextStep": "createPubSubnet"
    },
    {
      "name": "createPubSubnet",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "CreateSubnet",
        "CidrBlock": "10.0.103.0/24",
        "AvailabilityZone": "us-west-2c",
        "VpcId": "{{ createVpc.vpcId }}"
      },
      "outputs":[
        {
          "Name": "pubSubnetId",
          "Selector": "$.Subnet.SubnetId",
          "Type": "String"
        }
      ],
      "nextStep": "createPubRtb"
    },
    {
      "name": "createPubRtb",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "CreateRouteTable",
        "VpcId": "{{ createVpc.vpcId }}"
      },
      "outputs": [
        {
          "Name": "pubRtbId",
          "Selector": "$.RouteTable.RouteTableId",
          "Type": "String"
        }
      ],
      "nextStep": "createIgw"
    },
    {
      "name": "createIgw",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "CreateInternetGateway"
      },
      "outputs": [
        {
          "Name": "igwId",
          "Selector": "$.InternetGateway.InternetGatewayId",
          "Type": "String"
        }
      ],
      "nextStep": "attachIgw"
```

```
    },
    {
      "name": "attachIgw",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "AttachInternetGateway",
        "InternetGatewayId": "{{ createIgw.igwId }}",
        "VpcId": "{{ createVpc.vpcId }}"
      },
      "nextStep": "allocateEip"
    },
    {
      "name": "allocateEip",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "AllocateAddress",
        "Domain": "vpc"
      },
      "outputs": [
        {
          "Name": "eipAllocationId",
          "Selector": "$.AllocationId",
          "Type": "String"
        }
      ],
      "nextStep": "createNatGw"
    },
    {
      "name": "createNatGw",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "CreateNatGateway",
        "AllocationId": "{{ allocateEip.eipAllocationId }}",
        "SubnetId": "{{ createPubSubnet.pubSubnetId }}"
      },
      "outputs":[
        {
          "Name": "natGwId",
          "Selector": "$.NatGateway.NatGatewayId",
          "Type": "String"
        }
      ],
      "nextStep": "verifyNatGwAvailable"
    },
    {
      "name": "verifyNatGwAvailable",
      "action": "aws:waitForAwsResourceProperty",
      "timeoutSeconds": 150,
      "inputs": {
        "Service": "ec2",
        "Api": "DescribeNatGateways",
        "NatGatewayIds": [
          "{{ createNatGw.natGwId }}"
        ],
        "PropertySelector": "$.NatGateways[0].State",
        "DesiredValues": [
          "available"
        ]
      },
      "nextStep": "createNatRoute"
```

```
    },
    {
      "name": "createNatRoute",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "CreateRoute",
        "DestinationCidrBlock": "0.0.0.0/0",
        "NatGatewayId": "{{ createNatGw.natGwId }}",
        "RouteTableId": "{{ getMainRtb.mainRtbId }}"
      },
      "nextStep": "createPubRoute"
    },
    {
      "name": "createPubRoute",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "CreateRoute",
        "DestinationCidrBlock": "0.0.0.0/0",
        "GatewayId": "{{ createIgw.igwId }}",
        "RouteTableId": "{{ createPubRtb.pubRtbId }}"
      },
      "nextStep": "setPubSubAssoc"
    },
    {
      "name": "setPubSubAssoc",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "AssociateRouteTable",
        "RouteTableId": "{{ createPubRtb.pubRtbId }}",
        "SubnetId": "{{ createPubSubnet.pubSubnetId }}"
      }
    },
    {
      "name": "createDhcpOptions",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "CreateDhcpOptions",
        "DhcpConfigurations": [
          {
            "Key": "domain-name-servers",
            "Values": ["10.0.100.50,10.0.101.50"]
          },
          {
            "Key": "domain-name",
            "Values": ["sample.com"]
          }
        ]
      },
      "outputs": [
        {
          "Name": "dhcpOptionsId",
          "Selector": "$.DhcpOptions.DhcpOptionsId",
          "Type": "String"
        }
      ],
      "nextStep": "createDCSubnet1"
    },
    {
```

```
      "name": "createDCSubnet1",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "CreateSubnet",
        "CidrBlock": "10.0.100.0/24",
        "AvailabilityZone": "us-west-2a",
        "VpcId": "{{ createVpc.vpcId }}"
      },
      "outputs": [
        {
          "Name": "firstSubnetId",
          "Selector": "$.Subnet.SubnetId",
          "Type": "String"
        }
      ],
      "nextStep": "createDCSubnet2"
    },
    {
      "name": "createDCSubnet2",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "CreateSubnet",
        "CidrBlock": "10.0.101.0/24",
        "AvailabilityZone": "us-west-2b",
        "VpcId": "{{ createVpc.vpcId }}"
      },
      "outputs": [
        {
          "Name": "secondSubnetId",
          "Selector": "$.Subnet.SubnetId",
          "Type": "String"
        }
      ],
      "nextStep": "createDCSecGroup"
    },
    {
      "name": "createDCSecGroup",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "CreateSecurityGroup",
        "GroupName": "SampleDCSecGroup",
        "Description": "Security Group for Example Domain Controllers",
        "VpcId": "{{ createVpc.vpcId }}"
      },
      "outputs": [
        {
          "Name": "dcSecGroupId",
          "Selector": "$.GroupId",
          "Type": "String"
        }
      ],
      "nextStep": "authIngressDCTraffic"
    },
    {
      "name": "authIngressDCTraffic",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "AuthorizeSecurityGroupIngress",
```

```
      "GroupId": "{{ createDCSecGroup.dcSecGroupId }}",
      "IpPermissions": [
        {
          "FromPort": -1,
          "IpProtocol": "-1",
          "IpRanges": [
            {
              "CidrIp": "0.0.0.0/0",
              "Description": "Allow all traffic between Domain Controllers"
            }
          ]
        }
      ]
    },
    "nextStep": "verifyInstanceProfile"
  },
  {
    "name": "verifyInstanceProfile",
    "action": "aws:waitForAwsResourceProperty",
    "maxAttempts": 5,
    "onFailure": "Abort",
    "inputs": {
      "Service": "iam",
      "Api": "ListInstanceProfilesForRole",
      "RoleName": "sampleSSMInstanceRole",
      "PropertySelector": "$.InstanceProfiles[0].Arn",
      "DesiredValues": [
        "{{ createSSMInstanceProfile.instanceProfileArn }}"
      ]
    },
    "nextStep": "iamEventualConsistency"
  },
  {
    "name": "iamEventualConsistency",
    "action": "aws:sleep",
    "inputs": {
      "Duration": "PT2M"
    },
    "nextStep": "launchDC1"
  },
  {
    "name": "launchDC1",
    "action": "aws:executeAwsApi",
    "onFailure": "Abort",
    "inputs": {
      "Service": "ec2",
      "Api": "RunInstances",
      "BlockDeviceMappings": [
        {
          "DeviceName": "/dev/sda1",
          "Ebs": {
            "DeleteOnTermination": true,
            "VolumeSize": 50,
            "VolumeType": "gp2"
          }
        },
        {
          "DeviceName": "xvdf",
          "Ebs": {
            "DeleteOnTermination": true,
            "VolumeSize": 100,
            "VolumeType": "gp2"
          }
        }
      ],
      "IamInstanceProfile": {
```

```
          "Arn": "{{ createSSMInstanceProfile.instanceProfileArn }}"
        },
        "ImageId": "{{ getLatestWindowsAmi.amiId }}",
        "InstanceType": "t2.micro",
        "MaxCount": 1,
        "MinCount": 1,
        "PrivateIpAddress": "10.0.100.50",
        "SecurityGroupIds": [
          "{{ createDCSecGroup.dcSecGroupId }}"
        ],
        "SubnetId": "{{ createDCSubnet1.firstSubnetId }}",
        "TagSpecifications": [
          {
            "ResourceType": "instance",
            "Tags": [
              {
                "Key": "Name",
                "Value": "SampleDC1"
              }
            ]
          }
        ]
      },
      "outputs": [
        {
          "Name": "pdcInstanceId",
          "Selector": "$.Instances[0].InstanceId",
          "Type": "String"
        }
      ],
      "nextStep": "launchDC2"
    },
    {
      "name": "launchDC2",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "RunInstances",
        "BlockDeviceMappings": [
          {
            "DeviceName": "/dev/sda1",
            "Ebs": {
              "DeleteOnTermination": true,
              "VolumeSize": 50,
              "VolumeType": "gp2"
            }
          },
          {
            "DeviceName": "xvdf",
            "Ebs": {
              "DeleteOnTermination": true,
              "VolumeSize": 100,
              "VolumeType": "gp2"
            }
          }
        ],
        "IamInstanceProfile": {
          "Arn": "{{ createSSMInstanceProfile.instanceProfileArn }}"
        },
        "ImageId": "{{ getLatestWindowsAmi.amiId }}",
        "InstanceType": "t2.micro",
        "MaxCount": 1,
        "MinCount": 1,
        "PrivateIpAddress": "10.0.101.50",
        "SecurityGroupIds": [
```

```
        "{{ createDCSecGroup.dcSecGroupId }}"
      ],
      "SubnetId": "{{ createDCSubnet2.secondSubnetId }}",
      "TagSpecifications": [
        {
          "ResourceType": "instance",
          "Tags": [
            {
              "Key": "Name",
              "Value": "SampleDC2"
            }
          ]
        }
      ]
    },
    "outputs": [
      {
        "Name": "adcInstanceId",
        "Selector": "$.Instances[0].InstanceId",
        "Type": "String"
      }
    ],
    "nextStep": "verifyDCInstanceState"
  },
  {
    "name": "verifyDCInstanceState",
    "action": "aws:waitForAwsResourceProperty",
    "inputs": {
      "Service": "ec2",
      "Api": "DescribeInstanceStatus",
      "IncludeAllInstances": true,
      "InstanceIds": [
        "{{ launchDC1.pdcInstanceId }}",
        "{{ launchDC2.adcInstanceId }}"
      ],
      "PropertySelector": "$.InstanceStatuses[0].InstanceState.Name",
      "DesiredValues": [
        "running"
      ]
    },
    "nextStep": "verifyInstancesOnlineSSM"
  },
  {
    "name": "verifyInstancesOnlineSSM",
    "action": "aws:waitForAwsResourceProperty",
    "timeoutSeconds": 600,
    "inputs": {
      "Service": "ssm",
      "Api": "DescribeInstanceInformation",
      "InstanceInformationFilterList": [
        {
          "key": "InstanceIds",
          "valueSet": [
            "{{ launchDC1.pdcInstanceId }}",
            "{{ launchDC2.adcInstanceId }}"
          ]
        }
      ],
      "PropertySelector": "$.InstanceInformationList[0].PingStatus",
      "DesiredValues": [
        "Online"
      ]
    },
    "nextStep": "installADRoles"
  },
  {
```

```
      "name": "installADRoles",
      "action": "aws:runCommand",
      "inputs": {
        "DocumentName": "AWS-RunPowerShellScript",
        "InstanceIds": [
          "{{ launchDC1.pdcInstanceId }}",
          "{{ launchDC2.adcInstanceId }}"
        ],
        "Parameters": {
          "commands": [
            "try {",
            "  Install-WindowsFeature -Name AD-Domain-Services -
IncludeManagementTools",
            "}",
            "catch {",
            "  Write-Error \"Failed to install ADDS Role.\"",
            "}"
          ]
        }
      },
      "nextStep": "setAdminPassword"
    },
    {
      "name": "setAdminPassword",
      "action": "aws:runCommand",
      "inputs": {
        "DocumentName": "AWS-RunPowerShellScript",
        "InstanceIds": [
          "{{ launchDC1.pdcInstanceId }}"
        ],
        "Parameters": {
          "commands": [
            "net user Administrator \"sampleAdminPass123!\""
          ]
        }
      },
      "nextStep": "createForest"
    },
    {
      "name": "createForest",
      "action": "aws:runCommand",
      "inputs": {
        "DocumentName": "AWS-RunPowerShellScript",
        "InstanceIds": [
          "{{ launchDC1.pdcInstanceId }}"
        ],
        "Parameters": {
          "commands": [
            "$dsrmPass = 'sample123!' | ConvertTo-SecureString -asPlainText -Force",
            "try {",
            "   Install-ADDSForest -DomainName \"sample.com\" -DomainMode 6 -
ForestMode 6 -InstallDNS -DatabasePath \"D:\\NTDS\" -SysvolPath \"D:\\SYSVOL\" -
SafeModeAdministratorPassword $dsrmPass -Force",
            "}",
            "catch {",
            "   Write-Error $_",
            "}",
            "try {",
            "   Add-DnsServerForwarder -IPAddress \"10.0.100.2\"",
            "}",
            "catch {",
            "   Write-Error $_",
            "}"
          ]
        }
      },
```

```
      "nextStep": "associateDhcpOptions"
    },
    {
      "name": "associateDhcpOptions",
      "action": "aws:executeAwsApi",
      "onFailure": "Abort",
      "inputs": {
        "Service": "ec2",
        "Api": "AssociateDhcpOptions",
        "DhcpOptionsId": "{{ createDhcpOptions.dhcpOptionsId }}",
        "VpcId": "{{ createVpc.vpcId }}"
      },
      "nextStep": "waitForADServices"
    },
    {
      "name": "waitForADServices",
      "action": "aws:sleep",
      "inputs": {
        "Duration": "PT1M"
      },
      "nextStep": "promoteADC"
    },
    {
      "name": "promoteADC",
      "action": "aws:runCommand",
      "inputs": {
        "DocumentName": "AWS-RunPowerShellScript",
        "InstanceIds": [
          "{{ launchDC2.adcInstanceId }}"
        ],
        "Parameters": {
          "commands": [
            "ipconfig /renew",
            "$dsrmPass = 'sample123!' | ConvertTo-SecureString -asPlainText -Force",
            "$domAdminUser = \"sample\\Administrator\"",
            "$domAdminPass = \"sampleAdminPass123!\" | ConvertTo-SecureString -
asPlainText -Force",
            "$domAdminCred = New-Object
 System.Management.Automation.PSCredential($domAdminUser,$domAdminPass)",
            "try {",
            "   Install-ADDSDomainController -DomainName \"sample.com\" -InstallDNS
 -DatabasePath \"D:\\NTDS\" -SysvolPath \"D:\\SYSVOL\" -SafeModeAdministratorPassword
 $dsrmPass -Credential $domAdminCred -Force",
            "}",
            "catch {",
            "   Write-Error $_",
            "}"
          ]
        }
      }
    }
  ]
}
```

Restaurer un volume racine à partir du dernier instantané

Le système d'exploitation d'un volume racine peut être corrompu pour diverses raisons. Par exemple, à la suite d'une opération d'application de correctifs, le démarrage des instances peut échouer en raison d'un noyau ou d'un registre corrompu. L'automatisation des tâches courantes de dépannage, comme la restauration d'un volume racine à partir du dernier instantané réalisé avant l'opération d'application de correctifs, peut réduire les temps d'arrêt et accélérer la résolution des problèmes.AWS Systems Manager Les actions Automation peuvent vous aider à y parvenir.

L'exemple de document AWS Systems Manager Automation suivant effectue ces actions.

- Il utilise l'action Automation `aws:executeAwsApi` pour récupérer les détails du volume racine de l'instance.
- Il utilise l'action Automation `aws:executeScript` pour récupérer le dernier instantané du volume racine.
- Il utilise l'action Automation `aws:branch` pour poursuivre l'exécution si un instantané est trouvé pour le volume racine.

YAML

```
---
description: Custom Automation Troubleshooting Sample
schemaVersion: '0.3'
assumeRole: "{{ AutomationAssumeRole }}"
parameters:
  AutomationAssumeRole:
    type: String
    description: "(Required) The ARN of the role that allows Automation to perform
      the actions on your behalf. If no role is specified, Systems Manager Automation
      uses your IAM permissions to execute this document."
    default: ''
  InstanceId:
      type: String
      description: "(Required) The Instance Id whose root EBS volume you want to
 restore the latest Snapshot."
      default: ''
mainSteps:
- name: getInstanceDetails
  action: aws:executeAwsApi
  onFailure: Abort
  inputs:
    Service: ec2
    Api: DescribeInstances
    InstanceIds:
    - "{{ InstanceId }}"
  outputs:
    - Name: availabilityZone
      Selector: "$.Reservations[0].Instances[0].Placement.AvailabilityZone"
      Type: String
    - Name: rootDeviceName
      Selector: "$.Reservations[0].Instances[0].RootDeviceName"
      Type: String
  nextStep: getRootVolumeId
- name: getRootVolumeId
  action: aws:executeAwsApi
  onFailure: Abort
  inputs:
    Service: ec2
    Api: DescribeVolumes
    Filters:
    -  Name: attachment.device
       Values: ["{{ getInstanceDetails.rootDeviceName }}"]
    -  Name: attachment.instance-id
       Values: ["{{ InstanceId }}"]
  outputs:
    - Name: rootVolumeId
      Selector: "$.Volumes[0].VolumeId"
      Type: String
  nextStep: getSnapshotsByStartTime
- name: getSnapshotsByStartTime
  action: aws:executeScript
```

```
  timeoutSeconds: 45
  onFailure: Abort
  inputs:
    Runtime: python3.6
    Handler: getSnapshotsByStartTime
    InputPayload:
      rootVolumeId : "{{ getRootVolumeId.rootVolumeId }}"
    Script: |-
      def getSnapshotsByStartTime(events,context):
        import boto3

        #Initialize client
        ec2 = boto3.client('ec2')
        rootVolumeId = events['rootVolumeId']
        snapshotsQuery = ec2.describe_snapshots(
          Filters=[
            {
              "Name": "volume-id",
              "Values": [rootVolumeId]
            }
          ]
        )
        if not snapshotsQuery['Snapshots']:
          noSnapshotFoundString = "NoSnapshotFound"
          return { 'noSnapshotFound' : noSnapshotFoundString }
        else:
          jsonSnapshots = snapshotsQuery['Snapshots']
          sortedSnapshots = sorted(jsonSnapshots, key=lambda k: k['StartTime'],
 reverse=True)
          latestSortedSnapshotId = sortedSnapshots[0]['SnapshotId']
          return { 'latestSnapshotId' : latestSortedSnapshotId }
  outputs:
  - Name: Payload
    Selector: $.Payload
    Type: StringMap
  - Name: latestSnapshotId
    Selector: $.Payload.latestSnapshotId
    Type: String
  - Name: noSnapshotFound
    Selector: $.Payload.noSnapshotFound
    Type: String
  nextStep: branchFromResults
- name: branchFromResults
  action: aws:branch
  onFailure: Abort
  inputs:
    Choices:
    - NextStep: createNewRootVolumeFromSnapshot
      Not:
        Variable: "{{ getSnapshotsByStartTime.noSnapshotFound }}"
        StringEquals: "NoSnapshotFound"
  isEnd: true
- name: createNewRootVolumeFromSnapshot
  action: aws:executeAwsApi
  onFailure: Abort
  inputs:
    Service: ec2
    Api: CreateVolume
    AvailabilityZone: "{{ getInstanceDetails.availabilityZone }}"
    SnapshotId: "{{ getSnapshotsByStartTime.latestSnapshotId }}"
  outputs:
    - Name: newRootVolumeId
      Selector: "$.VolumeId"
      Type: String
  nextStep: stopInstance
- name: stopInstance
```

```
  action: aws:executeAwsApi
  onFailure: Abort
  inputs:
    Service: ec2
    Api: StopInstances
    InstanceIds:
    - "{{ InstanceId }}"
  nextStep: verifyVolumeAvailability
- name: verifyVolumeAvailability
  action: aws:waitForAwsResourceProperty
  timeoutSeconds: 120
  inputs:
    Service: ec2
    Api: DescribeVolumes
    VolumeIds:
    - "{{ createNewRootVolumeFromSnapshot.newRootVolumeId }}"
    PropertySelector: "$.Volumes[0].State"
    DesiredValues:
    - "available"
  nextStep: verifyInstanceStopped
- name: verifyInstanceStopped
  action: aws:waitForAwsResourceProperty
  timeoutSeconds: 120
  inputs:
    Service: ec2
    Api: DescribeInstances
    InstanceIds:
    - "{{ InstanceId }}"
    PropertySelector: "$.Reservations[0].Instances[0].State.Name"
    DesiredValues:
    - "stopped"
  nextStep: detachRootVolume
- name: detachRootVolume
  action: aws:executeAwsApi
  onFailure: Abort
  inputs:
    Service: ec2
    Api: DetachVolume
    VolumeId: "{{ getRootVolumeId.rootVolumeId }}"
  nextStep: verifyRootVolumeDetached
- name: verifyRootVolumeDetached
  action: aws:waitForAwsResourceProperty
  timeoutSeconds: 30
  inputs:
    Service: ec2
    Api: DescribeVolumes
    VolumeIds:
    - "{{ getRootVolumeId.rootVolumeId }}"
    PropertySelector: "$.Volumes[0].State"
    DesiredValues:
    - "available"
  nextStep: attachNewRootVolume
- name: attachNewRootVolume
  action: aws:executeAwsApi
  onFailure: Abort
  inputs:
    Service: ec2
    Api: AttachVolume
    Device: "{{ getInstanceDetails.rootDeviceName }}"
    InstanceId: "{{ InstanceId }}"
    VolumeId: "{{ createNewRootVolumeFromSnapshot.newRootVolumeId }}"
  nextStep: verifyNewRootVolumeAttached
- name: verifyNewRootVolumeAttached
  action: aws:waitForAwsResourceProperty
  timeoutSeconds: 30
  inputs:
```

```
    Service: ec2
    Api: DescribeVolumes
    VolumeIds:
    - "{{ createNewRootVolumeFromSnapshot.newRootVolumeId }}"
    PropertySelector: "$.Volumes[0].Attachments[0].State"
    DesiredValues:
    - "attached"
  nextStep: startInstance
- name: startInstance
  action: aws:executeAwsApi
  onFailure: Abort
  inputs:
    Service: ec2
    Api: StartInstances
    InstanceIds:
    - "{{ InstanceId }}"
```

JSON

```
{
   "description": "Custom Automation Troubleshooting Sample",
   "schemaVersion": "0.3",
   "assumeRole": "{{ AutomationAssumeRole }}",
   "parameters": {
      "AutomationAssumeRole": {
         "type": "String",
         "description": "(Required) The ARN of the role that allows Automation
 to perform the actions on your behalf. If no role is specified, Systems Manager
 Automation uses your IAM permissions to run this document.",
         "default": ""
      },
      "InstanceId": {
         "type": "String",
         "description": "(Required) The Instance Id whose root EBS volume you want to
 restore the latest Snapshot.",
         "default": ""
      }
   },
   "mainSteps": [
      {
         "name": "getInstanceDetails",
         "action": "aws:executeAwsApi",
         "onFailure": "Abort",
         "inputs": {
            "Service": "ec2",
            "Api": "DescribeInstances",
            "InstanceIds": [
               "{{ InstanceId }}"
            ]
         },
         "outputs": [
            {
               "Name": "availabilityZone",
               "Selector": "$.Reservations[0].Instances[0].Placement.AvailabilityZone",
               "Type": "String"
            },
            {
               "Name": "rootDeviceName",
               "Selector": "$.Reservations[0].Instances[0].RootDeviceName",
               "Type": "String"
            }
         ],
         "nextStep": "getRootVolumeId"
      },
```

```
    {
       "name": "getRootVolumeId",
       "action": "aws:executeAwsApi",
       "onFailure": "Abort",
       "inputs": {
          "Service": "ec2",
          "Api": "DescribeVolumes",
          "Filters": [
             {
                "Name": "attachment.device",
                "Values": [
                   "{{ getInstanceDetails.rootDeviceName }}"
                ]
             },
             {
                "Name": "attachment.instance-id",
                "Values": [
                   "{{ InstanceId }}"
                ]
             }
          ]
       },
       "outputs": [
          {
             "Name": "rootVolumeId",
             "Selector": "$.Volumes[0].VolumeId",
             "Type": "String"
          }
       ],
       "nextStep": "getSnapshotsByStartTime"
    },
    {
       "name": "getSnapshotsByStartTime",
       "action": "aws:executeScript",
       "timeoutSeconds": 45,
       "onFailure": "Continue",
       "inputs": {
          "Runtime": "python3.6",
          "Handler": "getSnapshotsByStartTime",
          "InputPayload": {
             "rootVolumeId": "{{ getRootVolumeId.rootVolumeId }}"
          },
          "Attachment": "getSnapshotsByStartTime.py"
       },
       "outputs": [
          {
             "Name": "Payload",
             "Selector": "$.Payload",
             "Type": "StringMap"
          },
          {
             "Name": "latestSnapshotId",
             "Selector": "$.Payload.latestSnapshotId",
             "Type": "String"
          },
          {
             "Name": "noSnapshotFound",
             "Selector": "$.Payload.noSnapshotFound",
             "Type": "String"
          }
       ],
       "nextStep": "branchFromResults"
    },
    {
       "name": "branchFromResults",
       "action": "aws:branch",
```

```
        "onFailure": "Abort",
        "inputs": {
           "Choices": [
              {
                 "NextStep": "createNewRootVolumeFromSnapshot",
                 "Not": {
                    "Variable": "{{ getSnapshotsByStartTime.noSnapshotFound }}",
                    "StringEquals": "NoSnapshotFound"
                 }
              }
           ]
        },
        "isEnd": true
     },
     {
        "name": "createNewRootVolumeFromSnapshot",
        "action": "aws:executeAwsApi",
        "onFailure": "Abort",
        "inputs": {
           "Service": "ec2",
           "Api": "CreateVolume",
           "AvailabilityZone": "{{ getInstanceDetails.availabilityZone }}",
           "SnapshotId": "{{ getSnapshotsByStartTime.latestSnapshotId }}"
        },
        "outputs": [
           {
              "Name": "newRootVolumeId",
              "Selector": "$.VolumeId",
              "Type": "String"
           }
        ],
        "nextStep": "stopInstance"
     },
     {
        "name": "stopInstance",
        "action": "aws:executeAwsApi",
        "onFailure": "Abort",
        "inputs": {
           "Service": "ec2",
           "Api": "StopInstances",
           "InstanceIds": [
              "{{ InstanceId }}"
           ]
        },
        "nextStep": "verifyVolumeAvailability"
     },
     {
        "name": "verifyVolumeAvailability",
        "action": "aws:waitForAwsResourceProperty",
        "timeoutSeconds": 120,
        "inputs": {
           "Service": "ec2",
           "Api": "DescribeVolumes",
           "VolumeIds": [
              "{{ createNewRootVolumeFromSnapshot.newRootVolumeId }}"
           ],
           "PropertySelector": "$.Volumes[0].State",
           "DesiredValues": [
              "available"
           ]
        },
        "nextStep": "verifyInstanceStopped"
     },
     {
        "name": "verifyInstanceStopped",
        "action": "aws:waitForAwsResourceProperty",
```

```
            "timeoutSeconds": 120,
            "inputs": {
               "Service": "ec2",
               "Api": "DescribeInstances",
               "InstanceIds": [
                  "{{ InstanceId }}"
               ],
               "PropertySelector": "$.Reservations[0].Instances[0].State.Name",
               "DesiredValues": [
                  "stopped"
               ]
            },
            "nextStep": "detachRootVolume"
         },
         {
            "name": "detachRootVolume",
            "action": "aws:executeAwsApi",
            "onFailure": "Abort",
            "inputs": {
               "Service": "ec2",
               "Api": "DetachVolume",
               "VolumeId": "{{ getRootVolumeId.rootVolumeId }}"
            },
            "nextStep": "verifyRootVolumeDetached"
         },
         {
            "name": "verifyRootVolumeDetached",
            "action": "aws:waitForAwsResourceProperty",
            "timeoutSeconds": 30,
            "inputs": {
               "Service": "ec2",
               "Api": "DescribeVolumes",
               "VolumeIds": [
                  "{{ getRootVolumeId.rootVolumeId }}"
               ],
               "PropertySelector": "$.Volumes[0].State",
               "DesiredValues": [
                  "available"
               ]
            },
            "nextStep": "attachNewRootVolume"
         },
         {
            "name": "attachNewRootVolume",
            "action": "aws:executeAwsApi",
            "onFailure": "Abort",
            "inputs": {
               "Service": "ec2",
               "Api": "AttachVolume",
               "Device": "{{ getInstanceDetails.rootDeviceName }}",
               "InstanceId": "{{ InstanceId }}",
               "VolumeId": "{{ createNewRootVolumeFromSnapshot.newRootVolumeId }}"
            },
            "nextStep": "verifyNewRootVolumeAttached"
         },
         {
            "name": "verifyNewRootVolumeAttached",
            "action": "aws:waitForAwsResourceProperty",
            "timeoutSeconds": 30,
            "inputs": {
               "Service": "ec2",
               "Api": "DescribeVolumes",
               "VolumeIds": [
                  "{{ createNewRootVolumeFromSnapshot.newRootVolumeId }}"
               ],
               "PropertySelector": "$.Volumes[0].Attachments[0].State",
```

```
            "DesiredValues": [
               "attached"
            ]
         },
         "nextStep": "startInstance"
      },
      {
         "name": "startInstance",
         "action": "aws:executeAwsApi",
         "onFailure": "Abort",
         "inputs": {
            "Service": "ec2",
            "Api": "StartInstances",
            "InstanceIds": [
               "{{ InstanceId }}"
            ]
         }
      }
   ],
   "files": {
        "getSnapshotsByStartTime.py": {
            "checksums": {
                "sha256": "sampleETagValue"
            }
        }
    }
}
```

Créer une AMI et une copie inter-régions

La création d'une Amazon Machine Image (AMI) d'une instance est un processus courant utilisé dans la sauvegarde et la restauration. Vous pouvez également choisir de copier une AMI dans une autre région dans le cadre d'une architecture de reprise après sinistre. L'automatisation des tâches courantes de maintenance peut réduire les temps d'arrêt si un problème nécessite un basculement.AWS Systems Manager Les actions Automation peuvent vous aider à y parvenir.

L'exemple de document AWS Systems Manager Automation suivant effectue ces actions.

- Il utilise l'action Automation `aws:executeAwsApi` pour créer une AMI.
- Il utilise l'action Automation `aws:waitForAwsResourceProperty` pour confirmer la disponibilité de l'AMI.
- Il utilise l'action Automation `aws:executeScript` pour copier l'AMI dans la région de destination.

YAML

```
---
description: Custom Automation Backup and Recovery Sample
schemaVersion: '0.3'
assumeRole: "{{ AutomationAssumeRole }}"
parameters:
  AutomationAssumeRole:
    type: String
    description: "(Required) The ARN of the role that allows Automation to perform
      the actions on your behalf. If no role is specified, Systems Manager Automation
      uses your IAM permissions to execute this document."
    default: ''
  InstanceId:
    type: String
    description: "(Required) The ID of the EC2 instance."
```

```
    default: ''
mainSteps:
- name: createImage
  action: aws:executeAwsApi
  onFailure: Abort
  inputs:
    Service: ec2
    Api: CreateImage
    InstanceId: "{{ InstanceId }}"
    Name: "Automation Image for {{ InstanceId }}"
    NoReboot: false
  outputs:
    - Name: newImageId
      Selector: "$.ImageId"
      Type: String
  nextStep: verifyImageAvailability
- name: verifyImageAvailability
  action: aws:waitForAwsResourceProperty
  timeoutSeconds: 600
  inputs:
    Service: ec2
    Api: DescribeImages
    ImageIds:
    - "{{ createImage.newImageId }}"
    PropertySelector: "$.Images[0].State"
    DesiredValues:
    - available
  nextStep: copyImage
- name: copyImage
  action: aws:executeScript
  timeoutSeconds: 45
  onFailure: Abort
  inputs:
    Runtime: python3.6
    Handler: crossRegionImageCopy
    InputPayload:
      newImageId : "{{ createImage.newImageId }}"
    Script: |-
      def crossRegionImageCopy(events,context):
        import boto3

        #Initialize client
        ec2 = boto3.client('ec2', region_name='us-east-1')
        newImageId = events['newImageId']

        ec2.copy_image(
          Name='DR Copy for ' + newImageId,
          SourceImageId=newImageId,
          SourceRegion='us-west-2'
        )
```

JSON

```
{
   "description": "Custom Automation Backup and Recovery Sample",
   "schemaVersion": "0.3",
   "assumeRole": "{{ AutomationAssumeRole }}",
   "parameters": {
      "AutomationAssumeRole": {
         "type": "String",
         "description": "(Required) The ARN of the role that allows Automation
 to perform\nthe actions on your behalf. If no role is specified, Systems Manager
 Automation\nuses your IAM permissions to run this document.",
         "default": ""
```

```
      },
      "InstanceId": {
         "type": "String",
         "description": "(Required) The ID of the EC2 instance.",
         "default": ""
      }
   },
   "mainSteps": [
      {
         "name": "createImage",
         "action": "aws:executeAwsApi",
         "onFailure": "Abort",
         "inputs": {
            "Service": "ec2",
            "Api": "CreateImage",
            "InstanceId": "{{ InstanceId }}",
            "Name": "Automation Image for {{ InstanceId }}",
            "NoReboot": false
         },
         "outputs": [
            {
               "Name": "newImageId",
               "Selector": "$.ImageId",
               "Type": "String"
            }
         ],
         "nextStep": "verifyImageAvailability"
      },
      {
         "name": "verifyImageAvailability",
         "action": "aws:waitForAwsResourceProperty",
         "timeoutSeconds": 600,
         "inputs": {
            "Service": "ec2",
            "Api": "DescribeImages",
            "ImageIds": [
               "{{ createImage.newImageId }}"
            ],
            "PropertySelector": "$.Images[0].State",
            "DesiredValues": [
               "available"
            ]
         },
         "nextStep": "copyImage"
      },
      {
         "name": "copyImage",
         "action": "aws:executeScript",
         "timeoutSeconds": 45,
         "onFailure": "Abort",
         "inputs": {
            "Runtime": "python3.6",
            "Handler": "crossRegionImageCopy",
            "InputPayload": {
               "newImageId": "{{ createImage.newImageId }}"
            },
            "Attachment": "crossRegionImageCopy.py"
         }
      }
   ],
   "files": {
        "crossRegionImageCopy.py": {
            "checksums": {
                "sha256": "sampleETagValue"
            }
        }
```

```
    }
}
```

Référence des documents Systems Manager Automation

Pour vous aider à démarrer rapidement, Systems Manager fournit des documents Automation prédéfinis. Ces documents sont tenus à jour par Amazon Web Services et AWS Support. La référence aux documents Automation décrit chacun des documents Automation prédéfinis fournis par AWS Systems Manager et AWS Support .

> **Important**
>
> Si vous exécutez un flux de travail d'automatisation qui appelle d'autres services à l'aide d'un rôle de service AWS Identity and Access Management (IAM), le rôle de service doit être configuré avec l'autorisation d'appeler ces services. Cette exigence s'applique à tous les documents AWS Automation (documents `AWS-*`) comme les documents `AWS-ConfigureS3BucketLogging`, `AWS-CreateDynamoDBBackup` et `AWS-RestartEC2Instance`, pour n'en citer que quelques-uns. Cette exigence s'applique également à tous les documents Automation personnalisés que vous créez qui appellent d'autres services AWS à l'aide d'actions qui appellent d'autres services. Par exemple, si vous utilisez les actions `aws:executeAwsApi`, `aws:createStack` ou `aws:copyImage`, vous devez configurer le rôle de service avec l'autorisation d'appeler ces services. Vous pouvez activer des autorisations à d'autres services AWS en ajoutant une stratégie IAM en ligne au rôle. Pour de plus amples informations, veuillez consulter (Facultatif) Ajoutez une stratégie en ligne Automation pour appeler d'autres services AWS (p. 311).

Référence sur les détails du document Automation

Cette section inclut des rubriques qui décrivent chacune des documents Systems Manager Automation appartenant à AWSet AWS Support. Chaque page fournit une explication des paramètres obligatoires et facultatifs que vous pouvez spécifier lors de l'utilisation du document. Chaque page répertorie également les étapes décrites dans le document et le résultat de l'exécution, le cas échéant.

Cette section n'inclut pas une page distincte pour les documents qui nécessitent une autorisation telles que le document AWS-CreateManagedLinuxInstanceWithApproval ou AWS-StopEC2InstanceWithApproval. Tout nom de document qui inclut WithApproval , signifie que le document inclut l'aws:approve – Suspendre une exécution pour approbation manuelle (p. 392)action. Cette action interrompt temporairement une exécution d'Automation jusqu'à ce que les mandataires désignés aient approuvé ou rejeté l'action. Une fois le nombre d'approbations requises atteint, l'exécution d'Automation reprend.

Pour de plus amples informations sur l'exécution de documents d'automatisation, veuillez consulter Exécution d'une automatisation simple (p. 314). Pour de plus amples informations sur l'exécution de documents d'automatisation sur plusieurs cibles, veuillez consulter Exécution d'automatisations qui utilisent des cibles et des contrôles du débit (p. 328).

Afficher le contenu du document Automation

Vous pouvez afficher le contenu des documents Automation dans la console Systems Manager.

Pour afficher le contenu du document Automation

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.

   -ou-

Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Documents dans le panneau de navigation.

3. Choisissez un document, puis choisissez View details (Afficher les détails).
4. Choisissez l'onglet Content (Contenu).

Rubriques

- AWSSupport-ActiverWindowsWithAmazonLicense (p. 513)
- AWS-ASGEnterStandby (p. 515)
- AWS-ASGExitStandby (p. 516)
- AWS-AttachEBSVolume (p. 517)
- AWS-AttachIAMToInstance (p. 518)
- AWSEC2-CloneInstanceAndUpgradeWindows (p. 519)
- AWSEC2-CloneInstanceAndUpgradeSQLServer (p. 522)
- AWSSupport-CollectEKSInstanceLogs (p. 524)
- AWS-ConfigureCloudWatchOnEC2Instance (p. 525)
- AWS-ConfigureS3BucketLogging (p. 526)
- AWS-ConfigureS3BucketVersioning (p. 527)
- Propriété STIG d'AWSEC2-Configure (p. 528)
- AWSSupport-Dépanneur de connectivité (p. 534)
- AWS-CopySnapshot (p. 537)
- AWS-CreateDynamoDBBackup (p. 538)
- AWS-CreateImage (p. 538)
- AWS-CreateJiraIssue (p. 539)
- AWS-CreateManagedLinuxInstance (p. 541)
- AWS-CreateManagedWindowsInstance (p. 542)
- AWS-CreateRdsSnapshot (p. 544)
- AWS-CreateServiceNowIncident (p. 545)
- AWS-CreateSnapshot (p. 547)
- AWS-DeleteCloudFormationStack (p. 547)
- AWS-DeleteDynamoDbBackup (p. 548)
- Sauvegardes AWS-DeleteDynamoDbTableBackups (p. 548)
- InstantanésVolumeSnapshots-DeleteEbs (p. 549)
- AWS-DeleteImage (p. 550)
- AWSConfigRemediation-DeleteUnusedIAMGroup (p. 551)
- AWS-DeleteSnapshot (p. 552)
- AWS-DetachEBSVolume (p. 553)
- AWS-DisablePublicAccessForSecurityGroup (p. 554)
- AWS-DisableS3BucketPublicReadWrite (p. 555)
- AWS-EnableCloudTrail (p. 555)
- AWSConfigRemediation-EnableEnhancedMonitoringOnRDSInstance (p. 556)
- AWS-EnableS3BucketEncryption (p. 557)
- AWSSupport-ExécuterEC2Rescue (p. 558)
- AWS-ExportOpsDataToS3 (p. 560)

- AWSSupport-AccorderPermissionsToIAMUser (p. 561)
- AWSSupport-ListEC2Ressources (p. 565)
- AWSSupport-ManageRDPSettings (p. 566)
- AWSSupport-ManageWindowsService (p. 568)
- Instance_correctif_AWS (p. 569)
- AWS-PatchInstanceWithRollback (p. 570)
- AWS-PublishSNSNotification (p. 572)
- AWS-RebootRdsInstance (p. 572)
- AWS-ReleaseElasticIP (p. 573)
- AWSSupport-ResetAccess (RéinitialiserAccès) (p. 574)
- AWS-ResizeInstance (p. 576)
- AWS-RestartEC2Instance (p. 576)
- AWS-RunCfnLint (p. 577)
- AWS-RunPacker (p. 579)
- AWSSupport-Envoyer le compartiment LogBundleToS3 (p. 580)
- AWS-SetupInventory (p. 581)
- AWSSupport-SetupIPMonitoringFromVPC (p. 584)
- AWS-SetupManagedInstance (p. 591)
- AWS-SetupManagedRoleOnEC2Instance (p. 592)
- AWSSupport-ShareRDSSnapshot (p. 593)
- AWSEC2-SQLServerDBRestore (p. 596)
- AWS-StartEC2Instance (p. 599)
- AWSSupport-StartEC2RescueWorkflow (p. 600)
- AWS-StartRdsInstance (p. 607)
- AWS-StopEC2Instance (p. 607)
- AWS-StopRdsInstance (p. 608)
- AWS-TerminateEC2Instance (p. 608)
- AWSSupport-TerminateIPMonitoringDeVPC (p. 609)
- AWSSupport-TroubleshootConnectivityToRDS (p. 611)
- AWSSupport-TroubleshootDirectoryTrust (p. 612)
- AWSSupport-TroubleshootRDP (p. 615)
- AWSSupport-TroubleshootS3PublicRead (p. 619)
- AWSSupport-DépannageSSH (p. 623)
- AWS-UpdateCloudFormationStack (p. 625)
- AWS-UpdateLinuxAmi (p. 625)
- AWS-UpdateWindowsAmi (p. 627)
- AWSSupport-UpgradeWindowsAWSDrivers (Mettre à niveauWindowsAWSDrivers) (p. 629)

AWSSupport-ActiverWindowsWithAmazonLicense

Description

Le document d'Automation AWSSupport -ActivateWindowsWithAmazonLicense active une instance EC2 pour Windows Server avec une licence fournie par Amazon. L'automatisation vérifie et configure les paramètres requis du système d'exploitation du service de gestion des clés et tente l'activation. Cela inclut

les routes du système d'exploitation vers les serveurs de gestion des clés d'Amazon et les paramètres du système d'exploitation du service de gestion des clés. La définition du paramètre `AllowOffline` sur `True` permet à l'automatisation de cibler avec succès les instances qui ne sont pas gérées par AWS Systems Manager, mais nécessite un arrêt et un démarrage de l'instance.

> **Note**
>
> Ce document ne peut pas être utilisé avec es instances Bring Your Own License (Licence à fournir) Windows Server. Pour plus d'informations sur l'utilisation de votre propre licence, consultez Licences Microsoft sur AWS.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows

Paramètres

- AllowOffline

  Type: String (chaîne)

  Valeurs valides : Vrai | Faux

  Par défaut : Faux

  Description : (Facultatif) Définissez-la sur True si vous autorisez la correction de l'activation Windows hors connexion au cas où le dépannage en ligne échouerait, ou si l'instance fournie n'est pas une instance gérée.

  > **Important**
  >
  > Le mode hors connexion nécessite l'arrêt, puis le redémarrage de l'instance EC2. Les données stockées sur les volumes de stockage d'instance seront perdues. L'adresse IP publique change si vous n'utilisez pas une adresse IP Elastic.

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.

- ForceActivation

  Type: String (chaîne)

  Valeurs valides : Vrai | Faux

  Par défaut : Faux

  Description : (Facultatif) Définissez ce paramètre sur True si vous souhaitez continuer même si Windows est déjà activé.

- InstanceId

  Type: String (chaîne)

  Description : (Obligatoire) ID de votre instance EC2 gérée pour Windows Server .
- SubnetId

  Type: String (chaîne)

  Par défaut : CreateNewVPC

  Description : (Facultatif) Offline only - ID de sous-réseau de l'EC2Rescueinstance utilisée pour effectuer le dépannage hors connexion. Utilisez SelectedInstanceSubnet pour utiliser le même sous-réseau que votre instance ou CreateNewVPC pour créer un VPC. IMPORTANT : Le sous-réseau doit être dans la même zone de disponibilité qu' InstanceId et doit autoriser l'accès aux points de terminaison SSM.

Autorisations IAM nécessaires

`AutomationAssumeRole` nécessite les actions suivantes pour exécuter correctement le document Automation.

Il est recommandé que l'instance EC2 recevant la commande dispose d'un rôle IAM avec la stratégie gérée AmazonSSMManagedInstanceCore Amazon attachée. Vous devez avoir au moins ssm:ExecuteAutomation et ssm:SendCommand pour exécuter Automation et envoyer la commande à l'instance, plus ssm:GetAutomationExecution pour pouvoir lire la sortie d'Automation. Pour la correction hors connexion, consultez les autorisations requises par AWSSupport -StartEC2RescueWorkflow.

Étapes de document

1. aws:assertAwsResourceProperty : vérifie que la plateforme de l'instance fournie est Windows.
2. aws:assertAwsResourceProperty : confirme que l'instance fournie est une instance gérée.
   a. (Correctif d'activation en ligne) Si l'instance d'entrée est une instance gérée, exécutez aws:runCommand pour exécuter le PowerShell script et tenter de corriger l'activation de Windows.
   b. (Correctif d'activation hors connexion) Si l'instance d'entrée n'est pas une instance gérée :
      i. aws:assertAwsResourceProperty : vérifie que l'AllowOfflineindicateur est défini sur True. Si tel est le cas, le correctif hors connexion démarre. Dans le cas contraire, le flux de travail se termine.
      ii. aws:executeAutomation : appelle -StartEC2RescueWorkflow avec le script de correction hors connexion de l'activation Windows.AWSSupport : le script s'appuie sur EC2Config ou EC2Launch en fonction de la version du système d'exploitation.
      iii. aws:executeAwsApi : lisez le résultat de AWSSupport -StartEC2RescueWorkflow.

Résultats

SortieactivateWindows

.SortiegetActivateWindowsOfflineResult

AWS-ASGEnterStandby

Description

Modifiez l'état de veille d'une instance EC2 dans un groupe Auto Scaling.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- InstanceId

  Type: String (chaîne)

  Description : (Obligatoire) ID d'une instance EC2 pour laquelle vous souhaitez modifier l'état de veille au sein d'un Auto Scaling groupe .
- LambdaRoleArn

  Type: String (chaîne)

  Description : (Facultatif) ARN du rôle qui autorise Lambda créé par Automation à effectuer les actions en votre nom. Si vous ne spécifiez pas cette valeur, un rôle transitoire est créé pour exécuter la fonction Lambda.

AWS-ASGExitStandby

Description

Modifiez l'état de veille d'une instance EC2 dans un groupe Auto Scaling.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.

- InstanceId

  Type: String (chaîne)

  Description : (Obligatoire) ID d'une instance EC2 pour laquelle vous souhaitez modifier l'état de veille au sein d'un Auto Scaling groupe .

- LambdaRoleArn

  Type: String (chaîne)

  Description : (Facultatif) ARN du rôle qui autorise Lambda créé par Automation à effectuer les actions en votre nom. Si vous ne spécifiez pas cette valeur, un rôle transitoire est créé pour exécuter la fonction Lambda.

AWS-AttachEBSVolume

Description

Attachez un volume Amazon Elastic Block Store (Amazon EBS) à une instance EC2.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.

- Appareil

  Type: String (chaîne)

  Description : (Obligatoire) Nom du périphérique (par exemple, /dev/sdh ou xvdh).

- InstanceId

  Type: String (chaîne)

  Description : (Obligatoire) ID de l'instance à laquelle vous souhaitez attacher le volume.

- VolumeId

Type: String (chaîne)

Description : (Obligatoire) ID du Amazon EBS volume. Le volume et l'instance doivent être dans la même zone de disponibilité.

AWS-AttachIAMToInstance

Description

Attachez un rôle AWS Identity and Access Management (IAM) à une instance gérée.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- ForceReplace

  Type: Booléen

  Description : (Facultatif) Indicateur permettant de spécifier s'il convient de remplacer le IAM profil existant ou non.

  Valeur par défaut : true
- InstanceId

  Type: String (chaîne)

  Description : (Obligatoire) ID de l'instance sur laquelle vous souhaitez attribuer un IAM rôle .
- RoleName

  Type: String (chaîne)

  Description : (Obligatoire) Nom du IAM rôle à ajouter à l'instance gérée.

Étapes de document

1. aws:executeAwsApi - Rechercher le profil dDescribeInstanceProfile'instance attaché à l'instance EC2.IAM
2. aws:branch - Vérifier le profil CheckInstanceProfileAssociations d'instance attaché à l'instance EC2.IAM

a. Si un profil d'instance IAM est attaché et que `ForceReplace` est défini sur `true` :
   i. aws:executeAwsApi - Dissocie le profil d'DisassociateIamInstanceProfileinstance de l'instance EC2.IAM
b. aws:executeAwsApi - Répertorie les profils d'instance pour le ListInstanceProfilesForRole rôle fourni.IAM
c. aws:branch - Vérifier si un profil d'instance est associé au CheckInstanceProfileCreated rôle fourni.IAM
   i. Si un profil d'instance est associé au rôle IAM :
      A. aws:executeAwsApi - AttachIAMProfileToInstance - Attacher le rôle de profil IAM d'instance à l'instance EC2.
   i. Si un aucun profil d'instance n'est associé au rôle IAM :
      A. aws:executeAwsApi - Créer un rôle de profil d'instance pour le CreateInstanceProfileForRole rôle spécifié.IAM
      B. aws:executeAwsApi - Attacher le rôle de profil d'instance au AddRoleToInstanceProfile rôle spécifié.IAM
      C. aws:executeAwsApi - Permet d'obtenir les données de profil d'instance pour le GetInstanceProfile rôle spécifié.IAM
      D. aws:executeAwsApi - AttachIAMProfileToInstanceWithRetry - Attacher le rôle de profil IAM d'instance à l'instance EC2.

Résultats

AttachIAMProfileToInstanceWithRetryID d'association

GetInstanceProfile.InstanceProfileName (NomProfilInstance)

GetInstanceProfileARN du profil d'instance

AttachIAMProfileToInstanceID d'association

ListInstanceProfilesForRole.InstanceProfileName (NomProfilInstance)

ListInstanceProfilesForRoleARN du profil d'instance

AWSEC2-CloneInstanceAndUpgradeWindows

Description

Créez une Amazon Machine Image (AMI) à partir d'une instance Windows Server 2008 R2, 2012 R2 ou 2016, puis mettez à niveau l'AMI vers Windows Server 2012 R2, 2016 ou 2019. Les chemins de mise à niveau pris en charge sont les suivants.

- Windows Server 2008 R2 vers Windows Server 2012 R2.
- Windows Server 2012 R2 vers Windows Server 2016.
- Windows Server 2012 R2 vers Windows Server 2019.
- Windows Server 2016 vers Windows Server 2019.

Pour mettre à niveau votre instance de Windows Server 2008 R2 vers Windows Server 2016 ou 2019, le document Automation effectue deux étapes. L'instance Windows Server 2008 R2 est mise à niveau vers Windows Server 2012 R2. Ensuite, l'instance Windows Server 2012 R2 est mise à niveau vers la version cible (Windows Server 2016 ou 2019).

L'opération de mise à niveau est un processus en plusieurs étapes qui peut prendre 2 heures. Automation crée une AMI à partir de l'instance, puis lance une instance temporaire à partir de l'AMI nouvellement créée

dans le `SubnetId` que vous spécifiez. Les groupes de sécurité associés à votre instance d'origine sont appliqués à l'instance temporaire. Le workflow Automation effectue ensuite une mise à niveau sur place vers la `TargetWindowsVersion` sur l'instance temporaire. Pour mettre à niveau votre instance Windows Server 2008 R2 vers Windows Server 2016 ou 2019, une mise à niveau sur place est effectuée deux fois, car la mise à niveau directe de Windows Server 2008 R2 vers Windows Server 2016 ou 2019 n'est pas prise en charge. Le flux de travail met également à jour ou installe les pilotes AWS requis par l'instance mise à niveau. Après la mise à niveau, le flux de travail crée une nouvelle AMI à partir de l'instance temporaire, puis met fin à l'instance temporaire.

Vous pouvez tester les fonctionnalités de l'application en lançant une instance de test à partir de l'AMI mise à niveau dans votre Amazon Virtual Private Cloud (Amazon VPC). Une fois que vous avez terminé le test et avant de procéder à une autre mise à niveau, planifiez les temps d'arrêt de l'application avant de passer complètement à l'AMI mise à niveau.

Exécution de cette automatisation (console)

Types de document

Automation

Propriétaire

Amazon

Plateformes

Éditions Standard et Datacenter de Windows Server 2008 R2, 2012 R2 et 2016

Conditions préalables requises

- Vérifiez que l'Agent SSM est installé sur votre instance. Pour plus d'informations, consultez la section Installation et configuration Agent SSM de sur les instances EC2 pourWindows Server (p. 100).
- Pour les instances qui sont jointes à un domaine Microsoft Active Directory, nous vous recommandons de spécifier un `SubnetId` qui n'a pas de connectivité à vos contrôleurs de domaine afin d'éviter les conflits de noms d'hôte.
- La `SubnetId` valeur spécifiée doit être un sous-réseau public avec l'option d'attribution automatique dIPv4'adresse publique définie sur true. Pour plus d'informations, consultez Modification de l'attribut d'IPv4adressage public pour votre sous-réseau dans le Amazon VPC Guide de l'utilisateur .
- Ce flux de travail Automation fonctionne uniquement avec des instances Windows Server 2008 R2, 2012 R2 et 2016.
- Ce flux de travail Automation fonctionne uniquement sur les instances EC2 avec un volume racine EBS non chiffré. Si l'instance spécifiée comporte un volume racine chiffré, le flux de travail Automation échoue.
- Configurez l'instance Windows Serveravec un profil d'instance AWS Identity and Access Management (IAM) qui fournit les autorisations requises pour Systems Manager. Pour plus d'informations, consultez la section Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34).
- Vérifiez que l'instance a 20 Go d'espace disque libre dans le disque de démarrage.
- Si l'instance n'utilise pas une licence Windows fournie par AWS, spécifiez un ID d'instantané Amazon EBS qui inclut le support d'installation de Windows Server 2012 R2. Pour ce faire :
  - Vérifiez que l'instance EC2 exécute actuellement Windows Server 2012 ou version ultérieure.
  - Créez un volume EBS de 6 Go dans la même zone de disponibilité que celle où l'instance est en cours d'exécution. Attachez le volume à l'instance. Montez-la, par exemple, en tant que lecteur D.
  - Cliquez avec le bouton droit de la souris sur le fichier ISO et montez-le sur une instance telle que le lecteur E.
  - Copiez le contenu du fichier ISO depuis le lecteur E:\ vers le lecteur D:\.

- Créez un instantané EBS de 6 Go comme à l'étape 2 ci-dessus.

Restrictions

Cette procédure Automation ne prend pas en charge la mise à niveau des contrôleurs de domaine Windows, les clusters ni les système d'exploitation de bureau Windows. Elle ne prend pas non plus en charge les instances EC2 pour Windows Server avec les rôles suivants installés.

- Hôte de session des services Bureau à distance (RDSH)
- Broker de connexion des services Bureau à distance (RDCB)
- Hôte de virtualisation des services Bureau à distance (RDVH)
- Accès web des services Bureau à distance (RDWA)

Parameters

- BYOLWindowsMediaSnapshotId

  Type: String (chaîne)

  Description : (Facultatif) ID de l'Amazon EBSinstantané à copier qui inclut le support d'installation de Windows Server 2012R2. Obligatoire uniquement si vous mettez à niveau une instance BYOL.
- IamInstanceProfile

  Type: String (chaîne)

  Description : (Obligatoire) Nom du profil d'IAMinstance qui permet Systems Manager à de gérer l'instance.
- InstanceId

  Type: String (chaîne)

  Description : (Obligatoire) Instance EC2 exécutant Windows Server 2008 R2 ou 2012 R2.
- KeepPreUpgradeImageBackUp

  Type: String (chaîne)

  Description : (Facultatif) Si ce paramètre est défini sur True, Automation ne supprime pas l'AMI créée à partir de l'instance EC2 avant la mise à niveau. S'il est défini sur True, vous devez supprimer l'AMI. Par défaut, l'AMI sera supprimée.
- SubnetId

  Type: String (chaîne)

  Description : (Obligatoire) Fournissez un ID de sous-réseau qui se trouve dans la même Amazon VPC instance que celle que vous souhaitez mettre à niveau. Vérifiez que le sous-réseau a une connexion sortante vers des services AWS, Amazon S3 et Microsoft (pour télécharger les correctifs).
- TargetWindowsVersion

  Type: String (chaîne)

  Description : (Obligatoire) Sélectionnez la version cible de Windows.

  Par défaut : 2012R2
- RebootInstanceBeforeTakingImage

  Type: String (chaîne)

Description : (Facultatif) Si ce paramètre est défini sur True, Automation redémarre l'instance avant de créer une AMI avant la mise à niveau. Par défaut, l'automatisation ne redémarre pas avant la mise à niveau.

AWSEC2-CloneInstanceAndUpgradeSQLServer

Description

Créez une AMI à partir d'une instance EC2 pour Windows Server exécutant SQL Server 2008 (ou version ultérieure), puis mettez à niveau l'AMI vers SQL Server 2016. La mise à niveau est un processus en plusieurs étapes qui peut prendre 2 heures. L'automatisation crée l'AMI à partir de l'instance, puis lance l'AMI nouvellement créée dans le sous-réseau que vous fournissez. Automation exécute ensuite une mise à niveau sur place de SQL Server 2008 (ou version ultérieure) vers SQL Server 2016. Après la mise à niveau, l'automatisation crée une AMI, puis met fin à l'instance mise à niveau.

Vous pouvez tester la fonctionnalité applicative en lançant la nouvelle AMI dans votre VPC. Une fois que vous avez terminé le test et avant de procéder à une autre mise à niveau, planifiez les temps d'arrêt de l'application avant de passer complètement à l'instance mise à niveau.

> **Note**
>
> Si vous souhaitez modifier le nom d'ordinateur de l'instance EC2 lancée à partir de la nouvelle AMI, veuillez consulter Renommer un ordinateur qui héberge une instance autonome de SQL Server.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows

Paramètres

Conditions préalables requises

- L'instance EC2 doit utiliser une version Windows Server qui soit Windows Server 2008 R2 (ou version ultérieure) et SQL Server 2008 (ou version ultérieure).
- Vérifiez que l'Agent SSM est installé sur votre instance. Pour plus d'informations, consultez la section Installation et configuration Agent SSM de sur les instances EC2 pourWindows Server (p. 100).
- Configurez l'instance pour utiliser un rôle de profil d'instance AWS Identity and Access Management (IAM). Pour plus d'informations, consultez la section Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34).
- Vérifiez que l'instance a 20 Go d'espace disque libre dans le disque de démarrage.
- Pour les instances qui utilisent une version SQL Server avec apport de sa propre licence (BYOL), les prérequis supplémentaires suivants s'appliquent :
  - Fournissez un ID d'instantané EBS qui inclut le support d'installation de SQL Server 2016. Pour ce faire :

1. Vérifiez que l'instance EC2 exécute Windows Server 2008 R2 ou version ultérieure.
2. Créez un volume EBS de 6 Go dans la même zone de disponibilité que celle où l'instance est en cours d'exécution. Attachez le volume à l'instance. Montez-la, par exemple, en tant que lecteur D.
3. Cliquez avec le bouton droit de la souris sur le fichier ISO et montez-le sur une instance telle que le lecteur E.
4. Copiez le contenu du fichier ISO depuis le lecteur E:\ vers le lecteur D:\.
5. Créez un instantané EBS du volume de 6 Go créé à l'étape 2.

Restrictions

- La mise à niveau peut uniquement être effectuée sur un serveur SQL à l'aide de l'authentification Windows.
- Vérifiez qu'il n'y a pas de correctifs et mises à jour de sécurité en attente sur les instances. Ouvrez le Panneau de configuration, puis choisissez Rechercher les mises à jour.
- Les déploiements SQL Server HA et le mode de mise en miroir ne sont pas pris en charge.

Parameters

- IamInstanceProfile

  Type: String (chaîne)

  Description : (Obligatoire) Profil IAM d'instance.
- InstanceId

  Type: String (chaîne)

  Description : (Obligatoire) Instance exécutant Windows Server 2008 R2 (ou version ultérieure) et SQL Server 2008 (ou version ultérieure).
- KeepPreUpgradeImageBackUp

  Type: String (chaîne)

  Description : (Facultatif) Si ce paramètre est défini sur True, Automation ne supprime pas l'AMI créée à partir de l'instance avant la mise à niveau. S'il est défini sur True, vous devez supprimer l'AMI. Par défaut, l'AMI sera supprimée.
- SubnetId

  Type: String (chaîne)

  Description : (Obligatoire) Fournissez un sous-réseau pour le processus de mise à niveau. Vérifiez que le sous-réseau a une connexion sortante vers des services AWS, Amazon S3 et Microsoft (pour télécharger les correctifs).
- SQLServerSnapshotId

  Type: String (chaîne)

  Description : (Conditionnel) ID d'instantané du support d'installation de SQL Server 2016. Ce paramètre est obligatoire pour les instances qui utilisent une version BYOL de SQL Server. Ce paramètre est facultatif pour les instances SQL Server avec licence incluse (instances lancées à l'aide d'une Amazon Machine Image (AMI) fournie par AWS pour Microsoft Server avec Microsoft SQL Server).
- RebootInstanceBeforeTakingImage

  Type: String (chaîne)

Description : (Facultatif) Si ce paramètre est défini sur True, Automation redémarre l'instance avant de créer une AMI avant la mise à niveau. Par défaut, l'automatisation ne redémarre pas avant la mise à niveau.

Résultats

AMIId: L'ID de l'AMI créée à partir de l'instance qui a été mise à niveau vers SQL Server 2016

AWSSupport-CollectEKSInstanceLogs

Description

Le document AWSSupport Automation -CollectEKSInstanceLogs rassemble les fichiers journaux liés au système d'exploitation et Amazon Elastic Kubernetes Service à (Amazon EKS) à partir d'une instance Amazon Elastic Compute Cloud (Amazon EC2) pour vous aider à résoudre les problèmes courants. Pendant que l'automatisation collecte les fichiers journaux associés, des modifications sont apportées à la structure du système de fichiers, y compris la création de répertoires temporaires, la copie des fichiers journaux dans les répertoires temporaires et la compression des fichiers journaux dans une archive. Cette activité peut entraîner une augmentation `CPUUtilization` de l'instance EC2. Pour plus d'informations sur `CPUUtilization` , consultez Métriques d'instance dans le Guide de l'utilisateur Amazon CloudWatch .

Si vous spécifiez une valeur pour le `LogDestination` paramètre, Automation évalue l'état de la stratégie du compartiment Amazon Simple Storage Service (Amazon S3) que vous spécifiez. Pour faciliter la sécurité des journaux collectés à partir de votre instance EC2, si l'état de la stratégie `isPublic` est défini sur `true` , ou si la liste de contrôle d'accès (ACL) accorde `READ|WRITE` des autorisations au groupe `All Users` Amazon S3 prédéfini, les journaux ne sont pas chargés. Pour plus d'informations sur les groupes Amazon S3 prédéfinis, consultez les groupes Amazon S3 prédéfinis dans le Amazon Simple Storage Service Manuel du développeur .

> **Note**
>
> Cette automatisation nécessite au moins 10 % d'espace disque disponible sur le volume racine Amazon Elastic Block Store (Amazon EBS) attaché à votre instance EC2. S'il n'y a pas suffisamment d'espace disque disponible sur le volume racine, l'automatisation s'arrête.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- EKSInstanceId

  Type: String (chaîne)

Description : (Obligatoire) ID de l'instance Amazon EKS EC2 à partir de laquelle vous souhaitez collecter les journaux. doit être installé sur l'instance que vous spécifiez.Agent SSM

- LogDestination

  Type: String (chaîne)

  Description : (Facultatif) Le compartiment S3 de votre compte dans lequel charger les journaux archivés.

Autorisations IAM nécessaires

`AutomationAssumeRole` nécessite les actions suivantes pour exécuter correctement le document Automation.

- `ssm:ExecuteAutomation`
- `ssm:GetAutomationExecution`
- `ssm:SendCommand`

Nous recommandons que l'instance EC2 recevant la commande dispose d'un rôle IAM avec la stratégie gérée AmazonSSMManagedInstanceCore Amazon attachée. Pour charger l'archive de journal dans le compartiment S3 que vous spécifiez dans le `LogDestination` paramètre, vous devez ajouter l'`s3:PutObject`autorisation.

Étapes de document

- aws:assertAwsResourceProperty - Confirme que le système d'exploitation de la valeur spécifiée dans le `EKSInstanceId` paramètre est Linux.
- aws:runCommand : rassemble le système d'exploitation et les fichiers journaux Amazon EKS associés, en les compressant en une archive dans le `/var/log` répertoire.
- aws:branch - Confirme si une valeur a été spécifiée pour le `LogDestination` paramètre.
- aws:runCommand : charge l'archive de journal dans le compartiment S3 que vous spécifiez dans le `LogDestination` paramètre.

AWS-ConfigureCloudWatchOnEC2Instance

Description

Activez ou désactivez la surveillance Amazon CloudWatch sur les instances gérées.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.

- InstanceId

  Type: String (chaîne)

  Description : (Obligatoire) ID de l'Amazon EC2instance sur laquelle vous souhaitez activer CloudWatch la surveillance.

- propriétés

  Type: String (chaîne)

  Description : (Facultatif) Ce paramètre n'est pas pris en charge. Répertorié ici pour des raisons de compatibilité descendante.

- status

  Valeurs valides : Activé | Désactivé

  Description : (Facultatif) Spécifie s'il convient d'activer ou de désactiver CloudWatch .

  Par défaut : Enabled

Étapes de document

configureCloudWatch- Configure CloudWatch sur l'Amazon EC2instance avec le statut donné.

Résultats

L'exécution de l'automatisation n'a pas de sortie.

AWS-ConfigureS3BucketLogging

Description

Activez la journalisation au niveau d'un compartiment Amazon Simple Storage Service (Amazon S3).

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.

- BucketName

  Type: String (chaîne)

  Description : (Obligatoire) Nom du Amazon S3 compartiment pour lequel vous souhaitez configurer la journalisation.
- GrantedPermission

  Type: String (chaîne)

  Valeurs valides : FULL_CONTROL | READ | WRITE

  Description : (Obligatoire) Autorisations de journalisation attribuées au bénéficiaire du compartiment.
- GranteeEmailAddress

  Type: String (chaîne)

  (Facultatif) Adresse e-mail du bénéficiaire.
- GranteeId

  Type: String (chaîne)

  Description : (Facultatif) ID d'utilisateur canonique du bénéficiaire.
- GranteeType

  Type: String (chaîne)

  Valeurs valides : CanonicalUser| | GroupeAmazonCustomerByEmail

  Description : (Obligatoire) Type de bénéficiaire.
- GranteeUri

  Type: String (chaîne)

  Description : (Facultatif) URI du groupe du bénéficiaire.
- TargetBucket

  Type: String (chaîne)

  Description : (Obligatoire) Spécifie le compartiment dans lequel vous Amazon S3 souhaitez stocker les journaux d'accès au serveur. Les journaux peuvent être fournis dans n'importe quel compartiment que vous possédez. Vous pouvez également configurer plusieurs compartiments pour diffuser leurs journaux vers le même compartiment cible. Dans ce cas, vous devez choisir un autre compartiment TargetPrefix source pour que les fichiers journaux livrés puissent se distinguer par clé.
- TargetPrefix

  Type: String (chaîne)

  Par défaut : /

  Description : (Facultatif) Spécifie un préfixe pour les clés sous lesquelles les fichiers journaux seront stockés.

AWS-ConfigureS3BucketVersioning

Description

Configurez la gestion des versions pour un compartiment Amazon Simple Storage Service (Amazon S3).

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- BucketName

  Type: String (chaîne)

  Description : (Obligatoire) Nom du compartiment S3 dont la configuration de chiffrement sera gérée.
- VersioningState

  Type: String (chaîne)

  Valeurs valides : Activé | Suspendu

  Par défaut : Enabled

  Description : (Facultatif) Appliqué au VersioningConfiguration .Status. Lorsque ce paramètre est défini sur « Enabled », ce processus permet la gestion des versions pour les objets du compartiment. Tous les objets ajoutés à ce compartiment reçoivent un ID de version unique. Lorsque ce paramètre est défini sur `Suspended` , ce processus désactive la gestion des versions pour les objets du compartiment. Tous les objets ajoutés au compartiment reçoivent l'ID de version `null`.

Propriété STIG d'AWSEC2-Configure

Les normes STIG (Security Technical Implementation Guides, guides d'implémentation de la sécurité technique) sont les normes de configuration créées par l'agence des systèmes d'information de la Défense américaine (DISA, Defense Information Systems Agency) pour les systèmes et les logiciels d'information. Pour rendre vos systèmes conformes aux normes STIG, vous devez installer, configurer et tester différents paramètres de sécurité.

Amazon EC2fournit un document SSM, AWSEC2-ConfigureSTIG, pour appliquer STIG à une instance afin de vous aider à créer rapidement des images conformes aux normes STIG. Le document STIG SSM recherche les erreurs de configuration et exécute un script de correction. Le document STIG SSM s'installe InstallRoot sous Windows AMIs à partir du Département de la Défense des États-Unis (DoD) pour installer et mettre à jour les DoD certificats, et supprimer les certificats inutiles pour maintenir la conformité STIG. Il n'y a pas de frais supplémentaires pour l'utilisation du document STIG SSM.

Vous pouvez choisir la catégorie de conformité STIG à appliquer.

Niveaux de conformité

- Élevé (Catégorie I)

  Correspond au risque le plus sérieux et inclut toute vulnérabilité pouvant entraîner une perte de confidentialité, de disponibilité ou d'intégrité.
- Moyenne (Catégorie II)

  Toute vulnérabilité pouvant entraîner une perte de confidentialité, de disponibilité ou d'intégrité. Ces risques pourraient être écartés.
- Faible (catégorie III)

  Toute vulnérabilité qui dégrade les mesures de protection contre la perte de confidentialité, de disponibilité ou d'intégrité.

Rubriques

- Paramètres STIG Windows (p. 529)
- Paramètres STIG Linux (p. 533)

Paramètres STIG Windows

Les composants STIG Windows sont conçus pour les serveurs autonomes et appliquent la stratégie de groupe locale. Vous pouvez appliquer des paramètres STIG faibles, moyens ou élevés.

Version STIG-Build-Windows-Low

Les paramètres STIG suivants ont été appliqués. Les paramètres qui n'ont pas été appliqués sont dus à des stratégies spécifiques à l'organisation, à des limitations techniques, à l'obligation pour les administrateurs de vérifier les paramètres du document, ou ils ne s'appliquent pas aux serveurs autonomes. Pour afficher la liste détaillée des paramètres qui ont été ou n'ont pas été appliqués, consultez la page STIG Win. Pour obtenir la liste complète des actuelsSTIGs, consultez la bibliothèque de STIGs documents . Pour obtenir des instructions sur la façon d'afficher la liste complète, consultez Comment afficher SRGs et STIGs .

- Version 5 de Windows Server 2019 STIG V1 :

  V-93233, V-93235, V-93237, V-93259, V-93261, V-93309, V-93409 et V-93541
- Version 12 de Windows Server 2016 STIG V1 :

  V-73499, V-73501, V-73503, V-73505, V-73543, V-73563 et V-73705
- Version 19 de Windows Server 2012R2 STIG V2 :

  V-1075, V-1128, V-1136, V-1151, V-1165, V-1172, V-1173, V-1174, V-3373, V-4108, V-4108, V-4111, V-4111, V-4111, V-4113, V-416716, V-6766, V-4445, V-11806, V-367, V267, V-367, V-367, V, V-366, V, V-366, 267, V, V, 267, V, V, 367, V, 367, V, 367, V, V, 367, V, V, V, V, 367, V, V, 367, V, V, 367, V, V, 367, V, V, V, V, V, V, 367, V262626, V, V, 32626262626, 3, V, 3, V, V, V,,, V, V, V, V, V, V,, V, V, V, V, V, V,, V, V, V, V,,, V,, V,, 2,, V,,, 32626262626262626, V, V, V, V, V, V, V, V, V, V, V, V, V, V, V, V, V, V, V, 2, V, V, 2
- Microsoft .NET Framework STIG 4.0 V1 version 9 :

  Aucun paramètre STIG appliqué.
- Pare-feu Windows utilisant STIG V1 version 7 :

  V-17425, V-17426, V-17427, V-17435, V-17436, V-17437, V-17445, V-17446 et V-17447
- Version 19 d'Internet Explorer 11 STIG V1 :

V-46477, V-46629 et V-97527

Version STIG-Build-Windows-Medium

Les paramètres STIG suivants ont été appliqués. Les paramètres qui n'ont pas été appliqués sont dus à des stratégies spécifiques à l'organisation, à des limitations techniques, à l'obligation pour les administrateurs de vérifier les paramètres du document, ou ils ne s'appliquent pas aux serveurs autonomes. Pour afficher la liste détaillée des paramètres qui ont été ou n'ont pas été appliqués, consultez la page STIG Win. Pour obtenir la liste complète des actuelsSTIGs, consultez la bibliothèque de STIGs documents . Pour obtenir des instructions sur la façon d'afficher la liste complète, consultez Comment afficher SRGs et STIGs .

- Version 5 de Windows Server 2019 STIG V1

  V-92961, V-92965, V-92967, V-92969, V-92971, V-92973, V-92979, V-92981, V-92983, V-92987, V-92989, V-93007, V-93009, V-93011, V-93015, V-93017, V-93045, V-93053, V-93055, V-93059, V-93063, V-93067, V-93069, V-93071, V-93073, V-93075, V-93079, V-93081, V-93083, V-93085, V-93087, V-93089, V-93091, V-93093, V-93095, V-93097, V-93099, V-93101, V-93103, V-93 à V-93107, V-93 à109, V-93111, V-93113, V-93115, V-93117, V-93119, V-93141, V-93143, V-93145, V-93151, V-93153, V-93155, V-93157, V-93159, V-93161, V-93163, V-93165, V-93167, V-93169, V-93171, V-93173, V-93175, V-93177, V-93179, V-93181, V-93197, V-93199, V-93233, V-93235, V-93237, V-93239, V-93243, V-93249, V-93251, V-93253, V-93255, V-93257, V-93259, V-93261, V-93263, V-93265, V-93267, V-93269, V-93285, V-93287, V-93293, V-93295, V-93297, V-93299, V-93305, V-93307, V-93309, V-93311, V-93313, V-93315, V-93317, V-93319, V-93335, V-93339, V-93367, V-93383, V-93385, V-93387, V-93389, V-93391, V-93393, V-93395, V-93397, V-93399, V-93401, V-93403, V-93405, V-93407, V-93409, V-93411, V-93413, V-93415, V-93421, V-93423, V-93425, V-93427, V-93429, V-93431, V-93433, V-93435, V-93445, V-93453, V-93455, V-93459, V-93463, V-93469, V-93471, V-93477, V-93479, V-93487, V-93489, V-93491, V-93493, V-93495, V-93497, V-93499, V-93501, V-93505, V-93517, V-93521, V-93523, V-93525, V-93527, V-93529, V-93533, V-93541, V-93547, V-93549, V-93551, V-93553, V-93555, V-93557, V-93559, V-93561, V-93563, V-93565, V à 102625
- Version 12 de Windows Server 2016 STIG V1

  V-73287, V-73291, V-73293, V-73295, V-73297, V-73299, V-73301, V-73309, V-73311, V-73313, V-73315, V-73317, V-73319, V-73321, V-73323, V-73413, V-73415, V-73419, V-73423, V-73427, V-73429, V-73431, V-73433, V-73443, V-73445, V-73447, V-73449, V-73451, V-73453, V-73455, V-73457, V-73459, V-73461, V-73463, V-73465, V-73467, V-73469, V-73471, V-73473, V-73475, V-73477, V-73479, V-73481, V-73483, V-73487, V-73489, V-73491, V-73493, V-73497, V-73499, V-73501, V-73503, V-73505, V-73507, V-73511, V-73521, V-73525, V-73527, V-73529, V-73531, V-73537, V-73539, V-73541, V-73543, V-73551, V-73553, V-73555, V-73557, V-73559, V-73561, V-73563, V-73565, V-73567, V-73569, V-73571, V-73573, V-73575, V-73577, V-73579, V-73581, V-73583, V-73587, V-73589, V-73591, V-73595, V-73597, V-73601, V-73603, V-73605, V-73607, V-73609, V-73627, V-73633, V-73635, V-73637, V-73639, V-73641, V-73643, V-73645, V-73653 V-73655, V-73657, V-73661, V-73663, V-73673, V-73677, V-73679, V-73681, V-73683, V-73685, V-73693, V-73695, V-73697, V-73699, V-73705, V-73707, V-73709, V-73711, V-73713, V-73715, V-73717, V-73719, V-73721, V-73727, V-73733, V-73739, V-73743, V-73745, V-73749, V-73753, V-73759, V-73763, V-73767, V-73771, V-73775, V-73781, V-73783, V-73785, V-73787, V-73789, V-73793, V-73795, V-73797, V-73799, V-7 à 3651, V-7 à 365 V-73807, V-73809, V-78123, V-78125, V-90359, V-90361, et V-102623
- Version 19 de Windows Server 2012R2 STIG V2

  V-1070, V-1075, V-1097, V-1098, V-1099, V-1104, V-1 à V-1107, V-1113, V-1128, V-1136, V-1141, V-1150, V-1151, V-1154, V-1155, V-1157, V-1162, V-1163, V-1164, V-1165, V-1166, V-1171, V-1172, V-1173, V-1174, V-3373, V-3374, V-3377, V-3378, V-3 à 35 V-3381, V-3382, V-3385, V-3449, V-3453, V-3454, V-3455, V-3456, V-3469, V-3470, V-3479, V-3 480, V-3481, V-3666, V-4108, V-4 110, V-4111, V-4112, V-4113, V-4116, V-4438, V-4442, V-4445, V-4447, V-4448, V-6831, V-6832, V-6833, V-6836,

V-11806, V-14228, V-14229, V-14230, V-14232, V-14234, V-14235, V-14236, V-14237, V-14239, V-14240, V-14241, V-14242, V-14243, V-14247, V-14249, V-14253, V-14259, V-14260, V-14261, V-14268, V-14269, V-14270, V-15666, V-15667, V-15672, V-15674, V-15682, V-15683, V-15684 V-15685, V-15686, V-15687, V-15696, V-15697, V-15698, V-15699, V-15700, V-15701, V-15702, V-15703, V-15704 V-15705, V-15706, V-15707, V-15718, V-15722, V-15727, V-15991, V-15997, V-15998, V-15999, V à 16 000, V-16008, V-16020, V-16021, V-16048, V-21 950, V-21951, V-21952, V-21953, V-21954, V-21955, V-21956, V-21960, V-21961, V-21963, V-21964, V-21965, V-21967, V-21969, V-21970, V-21971, V-21980, V-26469, V-26470, V-26472, V-26473, V-26474, V-26478, V-26480, V-26481, V-26482, V-26483, V-26484, V-26485, V-26486, V-26487, V-26488, V-26489, V-26490, V-26492, V-26493, V-26494, V-26496, V-26498, V-26499, V-26500, V-26504, V-26506, V-26529, V-26530, V-26533, V-26535, V-26537, V-26538, V-26539, V-26540, V-26541, V-26542, V-26543, V-26546, V-26547, V-26548, V-26549, V-26 à 255, V-26551, V-26552, V-26553, V-26555, V-26557, V-26558, V-26575, V-26576, V-26577, V-26578, V-26579, V-26580, V-26581, V-26582, V-26600, V-26604, V-26605, V-26606, V-2 50004 V-32272, V-32274, V-36656, V-36657, V-36667, V-36668, V-36673, V-36677, V-36678, V-36679, V-36680, V-36681, V-36684, V-36687, V-36696, V-36697, V-36698, V-36700, V-36707, V-36708, V-36709, V-36713, V-36714, V-36719, V-36720, V-36773, V-36776, V-36777, V-40200, V-40202, V à 7204, V-40206, V à 3237, V-43238, V-43239, V-43240, V-43241, V-43245, V-57633, V-57639, V-72753, V-73519, V-73523, V-73805, V-78057, V-78059, V-78061, V-78063, V-80473, V-80475, V-80477, et V-102619

- Version 9 de Microsoft .NET Framework STIG 4.0 V1

  V-81495

- Pare-feu Windows utilisant STIG V1 édition 7

  V-17415, V-17416, V-17417, V-17419, V-17425, V-17426, V-17427, V-17429, V-17435, V-17436, V-17437, V-17439, V-17445, V-17446 et V-17447

- Version 19 d'Internet Explorer 11 STIG

  V-46473, V-46475, V-46477, V-46481, V-46483, V-46501, V-46507, V-46509, V-46511, V-46513, V-46515, V-46517, V-46521, V-46523, V-46525, V-46543, V-46545, V-46547, V-46549, V-46553, V-46555, V-46573, V-46575, V-46577, V-46579, V-46581, V-46583, V-46587, V-46589, V-46591, V-46593, V-46597, V-46599, V-46601, V-46603, V-46605, V-46607, V-46609, V-46615, V-46617, V-46619, V-46621, V-46625, V-46629, V-46633, V-46635, V-46637, V-46639, V-46641, V-46643, V-46645, V-46647, V-46649, V-46653, V-46663, V-46665, V-46669, V-46681, V-46685, V-46689, V-46691, V-46693, V-46695, V-46701, V-46705, V-46709, V-46711, V-46713, V-46715, V-46717, V-46719, V-46721, V-46723, V-46725, V-46727, V-46729, V-46731, V-46733, V-46779, V-46781, V-46787, V-46789, V-46791, V-46797, V-46799, V-46801, V-46807, V-46811, V-46815, V-46819, V-46829, V-46841, V-46847, V-46849, V-46853, V-46857, V-46859, V-46861, V-46865, V-46869, V-46879, V-46883, V-46885, V-46889, V-46893, V-46895, V-46897, V-46903, V-46907, V-46921, V-46927, V-46939, V-46975, V-46981, V-46987, V-46995, V-46997, V-46999, V-47003, V-47005, V-47009, V-64711, V-64713, V-64715, V-64717, V-64719, V-64721, V-64723, V-64725, V-64729, V-72757, V-72759, V-72761, V-72763, V-75169, V-75 à 171, et V-97527

Version STIG-Build-Windows-High

Les paramètres STIG suivants ont été appliqués. Les paramètres qui n'ont pas été appliqués sont dus à des stratégies spécifiques à l'organisation, à des limitations techniques, à l'obligation pour les administrateurs de vérifier les paramètres du document, ou ils ne s'appliquent pas aux serveurs autonomes. Pour afficher une liste détaillée des paramètres qui ont été ou n'ont pas été appliqués, consultez la page STIG Win. Pour obtenir la liste complète des actuelsSTIGs, consultez la bibliothèque STIGs de documents . Pour obtenir des instructions sur la façon d'afficher la liste complète, consultez Comment afficher SRGs et STIGs .

- Version 5 de Windows Server 2019 STIG V1

  V-92961, V-92965, V-92967, V-92969, V-92971, V-92973, V-92979, V-92981, V-92983, V-92987, V-92989, V-93007, V-93009, V-93011, V-93015, V-93017, V-93045, V-93053, V-93055, V-93059,

V-93063, V-93065, V-93067, V-93069, V-93071, V-93073, V-93075, V-93079, V-93081, V-93083, V-93085, V-93087, V-93089, V-93091, V-93093, V-93095, V-93097, V-93099, V-93101, V-93103, V-93 à V-93107, V-93 à109, V-93111, V-93113, V-93115, V-93117, V-93119, V-93141, V-93143, V-93145, V-93151, V-93153, V-93155, V-93157, V-93159, V-93161, V-93163, V-93165, V-93167, V-93169, V-93171, V-93173, V-93175, V-93177, V-93179, V-93181, V-93197, V-93199, V-93215, V-93233, V-93235, V-93237, V-93239, V-93243, V-93249, V-93251, V-93253, V-93255, V-93257, V-93259, V-93261, V-93263, V-93265, V-93267, V-93269, V-93279, V-93285, V-93287, V-93289, V-93291, V-93293, V-93295, V-93297, V-93299, V-93 à 30 V-93305, V-93307, V-93309, V-93311, V-93313, V-93315, V-93317, V-93319, V-93335, V-93339, V-93367, V-93373, V-93375, V-93377, V-93383, V-93385, V-93387, V-93389, V-93391, V-93393, V-93395, V-93397, V-93399, V-93401, V-93403, V-93405, V-93407, V-93409, V-93411, V-93413, V-93415, V-93421, V-93423, V-93425, V-93427, V-93429, V-93431, V-93433, V-93435, V-93445, V-93453, V-93455, V-93459, V-93463, V-93465, V-93467, V-93469, V-93471, V-93477, V-93479, V-93487, V-93489, V-93491, V-93493, V-93495, V-93497, V-93499, V-93501, V-93503, V-93505, V-93507, V-93517, V-93521, V-93523, V-93525, V-93527, V-93529, V-93533, V-93537, V-93539, V-93541, V-93547, V-93549, V-93551, V-93553, V-93555, V-93557, V-93559, V-93561, V-93563, V-93565, et V-102625.

- Version 12 de Windows Server 2016 STIG V1

  V-73287, V-73291, V-73293, V-73295, V-73297, V-73299, V-73301, V-73309, V-73311, V-73313, V-73315, V-73317, V-73319, V-73321, V-73323, V-73325, V-73413, V-73415, V-73419, V-73423, V-73427, V-73429, V-73431, V-73433, V-73443, V-73445, V-73447, V-73449, V-73451, V-73453, V-73455, V-73457, V-73459, V-73461, V-73463, V-73465, V-73467, V-73469, V-73471, V-73473, V-73475, V-73477, V-73479, V-73481, V-73483, V-73487, V-73489, V-73491, V-73493, V-73497, V-73499, V-73501, V-73503, V-73505, V-73507, V-73511, V-73521, V-73525, V-73527, V-73529, V-73531, V-73537, V-73539, V-73541, V-73543, V-73545, V-73547, V-73549, V-73551, V-73553, V-73555, V-73557, V-73559, V-73561, V-73563, V-73565, V-73567, V-73569, V-73571, V-73573, V-73575, V-73577, V-73579, V-73581, V-73583, V-73585, V-73587, V-73589, V-73591, V-73593, V-73595, V-73597, V-73599, V-73601, V-73603, V-73605, V-73607, V-73609, V-73621, V-73627, V-73633, V-73635, V-73637, V-73639, V-73641, V-73643, V-73645, V-73653 V-73655, V-73657, V-73661, V-73663, V-73665, V-73667, V-73669, V-73673, V-73675, V-73677, V-73679, V-73681, V-73683, V-73685, V-73687, V-73691, V-73693, V-73695, V-73697, V-73699, V-73705, V-73707, V-73709, V-73711, V-73713, V-73715, V-73717, V-73719, V-73721, V-73727, V-73733, V-73739, V-73743, V-73745, V-73749, V-73753, V-73755, V-73759, V-73763, V-73767, V-73771, V-73775, V-73781, V-73783, V-73785, V-73787, V-73789, V-73793, V-73795, V-73797, V-73799, V-7 à 3651, V-7 à 365 V-73807, V-73809, V-78123, V-78125, V-90359, V-90361, et V-102623.

- Version 19 de Windows Server 2012R2 STIG V2

  V-1070, V-1075, V-1093, V-1097, V-1098, V-1099, V-1102, V-1104, V-1 à V-1107, V-1113, V-1128, V-1136, V-1141, V-1150, V-1151, V-1153, V-1154, V-1155, V-1157, V-1162, V-1163, V-1164, V-1165, V-1166, V-1171, V-1172, V-1173, V-1174, V-2372, V-2374, V-3337, V-3339, V-3343, V-3344, V-3373, V-3374, V-3377, V-3378, V-3379, V-3 à 35 V-3381, V-3382, V-3385, V-3449, V-3453, V-3454, V-3455, V-3456, V-3469, V-3470, V-3479, V-3 480, V-3481, V-3666, V-4108, V-4 110, V-4111, V-4112, V-4113, V-4116, V-4438, V-4442, V-4443, V-4445, V-4447, V-4448, V-6831, V-6832, V-6833, V-6834, V-6836, V-11806, V-14228, V-14229, V-14230, V-14232, V-14234, V-14235, V-14236, V-14237, V-14239, V-14240, V-14241, V-14242, V-14243, V-14247, V-14249, V-14253, V-14259, V-14260, V-14261, V-14268, V-14269, V-14270, V-15666, V-15667, V-15672, V-15674, V-15682, V-15683, V-15684 V-15685, V-15686, V-15687, V-15696, V-15697, V-15698, V-15699, V-15700, V-15701, V-15702, V-15703, V-15704 V-15705, V-15706, V-15707, V-15718, V-15722, V-15727, V-15991, V-15997, V-15998, V-15999, V à 16 000, V-16008, V-16020, V-16021, V-16048, V-18010, V-21 950, V-21951, V-21952, V-21953, V-21954, V-21955, V-21956, V-21960, V-21961, V-21963, V-21964, V-21965, V-21967, V-21969, V-21970, V-21971, V-21973, V-21980, V-22692, V-26070, V-26283, V-26469, V-26470, V-26472, V-26473, V-26474, V-26478, V-26479, V-26480, V-26481, V-26482, V-26483, V-26484, V-26485, V-26486, V-26487, V-26488, V-26489, V-26490, V-26492, V-26493, V-26494, V-26496, V-26498, V-26499, V-26500, V-26504, V-26506, V-26529, V-26530, V-26533, V-26535, V-26537, V-26538, V-26539, V-26540, V-26541, V-26542, V-26543, V-26546, V-26547, V-26548, V-26549, V-26 à 255, V-26551, V-26552, V-26553, V-26555, V-26557, V-26558, V-26575, V-26576, V-26577, V-26578, V-26579, V-26580, V-26581, V-26582, V-26600, V-26604, V-26605, V-26606, V-2

50004 V-32272, V-32274, V-34974, V-36656, V-36657, V-36667, V-36668, V-36673, V-36677, V-36678, V-36679, V-36680, V-36681, V-36684, V-36687, V-36696, V-36697, V-36698, V-36700, V-36707, V-36708, V-36709, V-36712, V-36713, V-36714, V-36718, V-36719, V-36720, V-36773, V-36776, V-36777, V-40200, V-40202, V à 7204, V-40206, V à 3237, V-43238, V-43239, V-43240, V-43241, V-43245, V-57633, V-57639, V-72753, V-73519, V-73523, V-73805, V-78057, V-78059, V-78061, V-78063, V-80473, V-80475, V-80477, et V-102619.

- Version 9 de Microsoft .NET Framework STIG 4.0 V1

  V-81495

- Pare-feu Windows utilisant STIG V1 édition 7

  V-17415, V-17416, V-17417, V-17418, V-17419, V-17425, V-17426, V-17427, V-17428, V-17429, V-17435, V-17436, V-17437, V-17438, V-17439, V-17445, V-17446 et V-1747

- Version 19 d'Internet Explorer 11 STIG

  V-46473, V-46475, V-46477, V-46481, V-46483, V-46501, V-46507, V-46509, V-46511, V-46513, V-46515, V-46517, V-46521, V-46523, V-46525, V-46543, V-46545, V-46547, V-46549, V-46553, V-46555, V-46573, V-46575, V-46577, V-46579, V-46581, V-46583, V-46587, V-46589, V-46591, V-46593, V-46597, V-46599, V-46601, V-46603, V-46605, V-46607, V-46609, V-46615, V-46617, V-46619, V-46621, V-46625, V-46629, V-46633, V-46635, V-46637, V-46639, V-46641, V-46643, V-46645, V-46647, V-46649, V-46653, V-46663, V-46665, V-46669, V-46681, V-46685, V-46689, V-46691, V-46693, V-46695, V-46701, V-46705, V-46709, V-46711, V-46713, V-46715, V-46717, V-46719, V-46721, V-46723, V-46725, V-46727, V-46729, V-46731, V-46733, V-46779, V-46781, V-46787, V-46789, V-46791, V-46797, V-46799, V-46801, V-46807, V-46811, V-46815, V-46819, V-46829, V-46841, V-46847, V-46849, V-46853, V-46857, V-46859, V-46861, V-46865, V-46869, V-46879, V-46883, V-46885, V-46889, V-46893, V-46895, V-46897, V-46903, V-46907, V-46921, V-46927, V-46939, V-46975, V-46981, V-46987, V-46995, V-46997, V-46999, V-47003, V-47005, V-47009, V-64711, V-64713, V-64715, V-64717, V-64719, V-64721, V-64723, V-64725, V-64729, V-72757, V-72759, V-72761, V-72763, V-75169, V-75 à 171, et V-97527

Paramètres STIG Linux

Les sections suivantes contiennent des informations sur les paramètres STIG Linux qui peuvent être appliqués à Amazon Linux 2, Red Hat Enterprise Linux 7 et CentOs 7. Vous pouvez appliquer des paramètres STIG faibles, moyens ou élevés.

Version STIG-Build-Linux-Low

Les paramètres STIG suivants ont été appliqués. Les paramètres qui n'ont pas été appliqués sont dus à des stratégies spécifiques à l'organisation, à des limitations techniques, à l'obligation pour les administrateurs de vérifier les paramètres du document, ou ils ne s'appliquent pas aux serveurs autonomes. Pour afficher la liste détaillée des paramètres qui ont été ou n'ont pas été appliqués, consultez RHEL7 V2 R8 STIG. Pour obtenir la liste complète des actuelsSTIGs, consultez la bibliothèque de STIGs documents . Pour obtenir des instructions sur la façon d'afficher la liste complète, consultez Comment afficher SRGs et STIGs .

Version 8 de RHEL 7 STIG V

V-71987, V-72217, V-72275

Version STIG-Build-Linux-Medium

Les paramètres STIG suivants ont été appliqués. Les paramètres qui n'ont pas été appliqués sont dus à des stratégies spécifiques à l'organisation, à des limitations techniques, à l'obligation pour les administrateurs de vérifier les paramètres du document, ou ils ne s'appliquent pas aux serveurs autonomes. Pour afficher la liste détaillée des paramètres qui ont été ou n'ont pas été appliqués, consultez RHEL7 V2 R8 STIG. Pour obtenir la liste complète des actuelsSTIGs, consultez la bibliothèque STIGs de

documents . Pour obtenir des instructions sur la façon d'afficher la liste complète, consultez Comment afficher SRGs et STIGs .

Version 8 de RHEL 7 STIG V

V-71903, V-71905, V-71907, V-71909, V-71911, V-71913, V-71915, V-71917, V-719 V-71921, V-71923, V-71925, V-71933, V-71935, V-71941, V-71943, V-71945, V-71951, V-71959, V-71983, V-71985, V-71987, V-71995, V-72013, V-72095, V-72097, V-72099, V-72101, V-72103, V-72 à V-72107, V-72109, V-72111, V-72113, V-72115, V-72117, V-72119, V-72121, V-72123, V-72125, V-72127, V-72129, V-72131, V-72133, V-72 135, V-72137, V-72139, V-72141, V-72145, V-72147, V-72149, V-72151, V-72153, V-72155, V-72157, V-72159, V-72161, V-72163, V-72165, V-72167, V-72171, V-72173, V-72175, V-72177, V-72179, V-72183, V-72185, V-72187, V-72189, V-72191, V-72197, V-72199, V-72203, V-72205, V-72207, V-72217, V-72221, V-72223, V-72233, V-72235, V-72237, V-72241, V-72243, V-72245, V-72247, V-72249, V-72253, V-72259, V-72261, V-72263, V-72265, V-72267, V-72275, V-72283, V-72285, V-72287, V-72289, V-72291, V-72293, V-72297, V-72307, V-72309, V-72319, V-72417, V-72433, V-73159, V-73165, V-73167, V-73171, V-73173, V-73175, V-77821, V-77825, V-78999, V-79001, V 81003, V-92251, V-92253

Version STIG-Build-Linux-High

Les paramètres STIG suivants ont été appliqués. Les paramètres qui n'ont pas été appliqués sont dus à des stratégies spécifiques à l'organisation, à des limitations techniques, à l'obligation pour les administrateurs de vérifier les paramètres du document, ou ils ne s'appliquent pas aux serveurs autonomes. Pour afficher la liste détaillée des paramètres qui ont été ou n'ont pas été appliqués, consultez RHEL7 V2 R8 STIG. Pour obtenir la liste complète des actuelsSTIGs, consultez la bibliothèque de STIGs documents . Pour obtenir des instructions sur la façon d'afficher la liste complète, consultez Comment afficher SRGs et STIGs .

Version 8 de RHEL 7 STIG V

V-71903, V-71905, V-71907, V-71909, V-71911, V-71913, V-71915, V-71917, V-719 V-71921, V-71923, V-71925, V-71933, V-71935, V-71939, V-71941, V-71943, V-71945, V-71951, V-71959, V-71967, V-71969, V-71977, V-71979, V-71983, V-71985, V-71987, V-71993, V-71995, V-72013, V-72077, V-72079, V-72095, V-72097, V-72099, V-72101, V-72103, V-72 à V-72107, V-72109, V-72111, V-72113, V-72115, V-72117, V-72119, V-72121, V-72123, V-72125, V-72127, V-72129, V-72131, V-72133, V-72 135, V-72137, V-72139, V-72141, V-72145, V-72147, V-72149, V-72151, V-72153, V-72155, V-72157, V-72159, V-72161, V-72163, V-72165, V-72167, V-72171, V-72173, V-72175, V-72177, V-72179, V-72183, V-72185, V-72187, V-72189, V-72191, V-72197, V-72199, V-72203, V-72205, V-72207, V-72217, V-72221, V-72223, V-72233, V-72235, V-72237, V-72241, V-72243, V-72245, V-72247, V-72249, V-72253, V-72259, V-72261, V-72263, V-72265, V-72267, V-72275, V-72283, V-72285, V-72287, V-72289, V-72291, V-72293, V-72297, V-72299, V-72301, V-72303, V-72307, V-72309, V-72319, V-72417, V-72433, V-73159, V-73165, V-73167, V-73171, V-73173, V-73175, V-77821, V-77825, V-78999, V-79001, V 81003, V-92251, V-92253

AWSSupport-Dépanneur de connectivité

Description

Le document AWSSupport -ConnectivityTroubleshooter Automation diagnostique les problèmes de connectivité entre les éléments suivants :

- AWSRessources au sein d'un Amazon Virtual Private Cloud (Amazon VPC)
- AWSDes ressources dans différentes Amazon VPC au sein de la même AWS région qui sont connectées à l'aide de l'appairage de VPC
- AWSRessources dans un Amazon VPC et une ressource Internet à l'aide d'une passerelle Internet
- AWSRessources dans un Amazon VPC et une ressource Internet à l'aide d'une passerelle NAT (Network Address Translation)

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- DestinationIP

  Type: String (chaîne)

  Description : (Obligatoire) Adresse de la ressource à laquelle vous souhaitez vous connecter.IPv4
- DestinationPort

  Type: String (chaîne)

  Par défaut : True

  Description : (Obligatoire) Numéro de port auquel vous souhaitez vous connecter sur la ressource de destination.
- DestinationVpc

  Type: String (chaîne)

  Par défaut : Toutes

  Description : (Facultatif) ID du vers lequel Amazon VPC vous souhaitez tester la connectivité.
- SourceIP

  Type: String (chaîne)

  Description : (Obligatoire) Adresse privée de la IPv4 ressource de votre à partir de laquelle AWS vous souhaitez tester la connectivité.Amazon VPC
- SourcePortRange

  Type: String (chaîne)

  Description : (Facultatif) La plage de ports utilisée par la AWS ressource dans votre à partir de laquelle Amazon VPC vous souhaitez tester la connectivité.
- SourceVpc

  Type: String (chaîne)

  Par défaut : Toutes

  Description : (Facultatif) ID du à partir duquel Amazon VPC vous souhaitez tester la connectivité.

Autorisations IAM nécessaires

`AutomationAssumeRole` nécessite les actions suivantes pour exécuter correctement le document Automation.

- `ec2:DescribeNatGateways`
- `ec2:DescribeNetworkAcls`
- `ec2:DescribeNetworkInterfaces`
- `ec2:DescribeRouteTables`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeVpcPeeringConnections`

Étapes de document

- aws:executeScript - Collecte des détails sur la AWS ressource que vous spécifiez dans le `SourceIP` paramètre.
- aws:executeScript : détermine la destination du trafic réseau à partir de la AWS ressource à l'aide des routes recueillies à l'étape précédente.
- aws:branch - Branches basées sur la destination du trafic réseau.
- aws:executeAwsApi - Collecte des détails sur la ressource de destination.
- aws:executeScript - Confirme que l'ID renvoyé pour la destination Amazon VPC correspond à la valeur spécifiée, le cas échéant, dans le `DestinationVpc` paramètre.
- aws:executeAwsApi - Collecte les règles de groupe de sécurité pour les ressources source et de destination.
- aws:executeScript : confirme si les règles de groupe de sécurité autorisent le trafic nécessaire entre les ressources source et de destination.
- aws:executeAwsApi - Collecte les listes de contrôle d'accès (NACL) réseau associées aux sous-réseaux pour les ressources source et de destination.
- aws:executeScript - Confirme si l' NACLs autorise le trafic nécessaire entre les ressources source et de destination.
- aws:executeScript : confirme si la source a une adresse IP publique associée à la ressource, si la destination de routage est une passerelle Internet.
- aws:executeAwsApi - Collecte les règles de groupe de sécurité pour la ressource source.
- aws:executeScript : confirme si les règles de groupe de sécurité autorisent le trafic nécessaire de la source vers la ressource de destination.
- aws:executeAwsApi - Collecte l' associé au sous-réseau de la ressource source.NACLs
- aws:executeScript : confirme si l' NACLs autorise le trafic nécessaire à partir de la ressource source.
- aws:executeAwsApi - Collecte des détails sur la passerelle NAT.
- aws:executeAwsApi - Collecte l' associé au sous-réseau de la passerelle NAT.NACLs
- aws:executeScript - Confirme si l' NACLs autorise le trafic nécessaire à partir du sous-réseau pour la passerelle NAT.
- aws:executeScript - Collecte les routes associées au sous-réseau de la passerelle NAT.
- aws:executeScript - Confirme si la passerelle NAT dispose d'une route vers une passerelle Internet.
- aws:executeAwsApi - Collecte des détails sur la connexion d'appairage de VPC.
- aws:executeScript - Confirme que les deux VPCs se trouvent dans la même région et que l'ID renvoyé pour le VPC de destination correspond à la valeur spécifiée, le cas échéant, dans le `DestinationVpc` paramètre.
- aws:executeAwsApi - Renvoie le sous-réseau de la ressource de destination.
- aws:executeScript - Collecte les routes associées au sous-réseau pour le VPC appairé.
- aws:executeScript - Confirme si le VPC appairé a une route vers la connexion d'appairage.

- aws:executeScript - Confirme si le trafic est autorisé à partir de la ressource source si la destination n'est pas prise en charge par l'automatisation.

AWS-CopySnapshot

Description

Copie un instantané à un instant dans le passé d'un volume Amazon Elastic Block Store (Amazon EBS). Vous pouvez copier l'instantané dans la même région AWS ou d'une région à une autre. Les copies des instantanés Amazon EBS chiffrés restent chiffrées. Les copies des instantanés non chiffrés restent non chiffrées. Pour copier un instantané chiffré qui a été partagé à partir d'un autre compte, vous devez disposer des autorisations pour la clé CMK du client AWS KMS utilisée pour chiffrer l'instantané. Les instantanés créés par l'action CopySnapshot ont un ID de volume arbitraire qui ne doit être utilisé en aucun cas.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- Description

  Type: String (chaîne)

  Description : (Facultatif) Description de l'Amazon EBSinstantané.
- SnapshotId

  Type: String (chaîne)

  Description : (Obligatoire) ID de l'Amazon EBSinstantané à copier.
- SourceRegion

  Type: String (chaîne)

  Description : (Obligatoire) La région où l'instantané source existe actuellement.

Étapes de document

copySnapshot- Copie un instantané d'un Amazon EBS volume .

Résultats

copySnapshot.SnapshotId - ID du nouvel instantané.

AWS-CreateDynamoDBBackup

Description

Créez une sauvegarde d'une table Amazon DynamoDB.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- BackupName

  Type: String (chaîne)

  Description : (Obligatoire) Nom de la sauvegarde à créer.
- LambdaAssumeRole

  Type: String (chaîne)

  Description : (Facultatif) ARN du rôle qui autorise Lambda créé par Automation à effectuer les actions en votre nom. Si vous ne spécifiez pas cette valeur, un rôle transitoire est créé pour exécuter la fonction Lambda.
- TableName

  Type: String (chaîne)

  Description : (Obligatoire) Nom de la DynamoDB table.

AWS-CreateImage

Description

Créez une Amazon Machine Image (AMI) à partir d'une instance EC2.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- InstanceId

  Type: String (chaîne)

  Description : (Obligatoire) ID de l'instance EC2.
- NoReboot

  Type: Booléen

  Description : (Facultatif) Ne redémarrez pas l'instance avant de créer l'image.

Autorisations IAM nécessaires

`AutomationAssumeRole` nécessite les actions suivantes pour exécuter correctement le document Automation.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateImage",
                "ec2:DescribeImages"
            ],
            "Resource": [
                "*"
            ]
        }
    ]
}
```

AWS-CreateJiraIssue

Description

Création d'un problème de Jira.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AssigneeName

  Type: String (chaîne)

  Description : (Facultatif) Nom d'utilisateur de la personne à laquelle le problème doit être attribué.
- DueDate

  Type: String (chaîne)

  Description : (Facultatif) Date d'échéance du problème au format aaaa-mm-jj.
- IssueDescription

  Type: String (chaîne)

  Description : (Obligatoire) Description détaillée du problème.
- IssueSummary

  Type: String (chaîne)

  Description : (Obligatoire) Un bref résumé du problème.
- IssueTypeName

  Type: String (chaîne)

  Description : (Obligatoire) Le nom du type de problème que vous souhaitez créer (par exemple, tâche, sous-tâche, bogue, etc.).
- JiraURL

  Type: String (chaîne)

  Description : (Obligatoire) URL de l'instance Jira.
- JiraUsername

  Type: String (chaîne)

  Description : (Obligatoire) Nom de l'utilisateur avec lequel le problème sera créé.
- PriorityName

  Type: String (chaîne)

  Description : (Facultatif) Nom de la priorité du problème.
- ProjectKey

  Type: String (chaîne)

  Description : (Obligatoire) Clé du projet dans lequel le problème doit être créé.
- SSMParameterName

  Type: String (chaîne)

Description : (Obligatoire) Nom d'un paramètre SSM chiffré contenant la clé d'API ou le mot de passe de l'utilisateur Jira.

Étapes de document

aws:createStack : créez CloudFormation une pile pour créer un rôle et une fonction Lambda IAM.

aws:invokeLambdaFunction - Appel de la fonction Lambda pour créer le problème Jira

aws:deleteStack : supprimez la CloudFormation pile créée.

Résultats

IssueId: ID du problème Jira nouvellement créé

AWS-CreateManagedLinuxInstance

Description

Créez une instance EC2 pour Linux configurée pour Systems Manager.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Linux

Paramètres

- AmiId

  Type: String (chaîne)

  Description : (Obligatoire) ID d'AMI à utiliser pour lancer l'instance.
- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- GroupName

  Type: String (chaîne)

  Par défaut : SSMSecurityGroupForLinuxInstances

  Description : (Obligatoire) Nom du groupe de sécurité à créer.
- InstanceType

  Type: String (chaîne)

Par défaut : t2.medium

Description : (Obligatoire) Type d'instance à lancer. La valeur par défaut est t2.medium.

- KeyPairName

  Type: String (chaîne)

  Description : (Obligatoire) Paire de clés à utiliser lors de la création de l'instance.

- RemoteAccessCidr

  Type: String (chaîne)

  Par défaut : 0.0.0.0/0

  Description : (Obligatoire) Crée un groupe de sécurité avec le port SSH (plage de ports 22) ouvert pour IPs spécifié par CIDR (la valeur par défaut est 0.0.0.0/0). Si le groupe de sécurité existe déjà, il ne sera pas modifié et les règles ne changeront pas.

- RoleName

  Type: String (chaîne)

  Par défaut : SSMManagedInstanceProfileRole

  Description : (Obligatoire) Nom du rôle à créer.

- StackName

  Type: String (chaîne)

  Par défaut : CreateManagedInstanceStack{{automation:EXECUTION_ID}}

  Description : (Facultatif) Spécifiez le nom de la pile utilisé par ce document

- SubnetId

  Type: String (chaîne)

  Par défaut : Valeur par défaut

  Description : (Obligatoire) Une nouvelle instance sera déployée dans ce sous-réseau ou dans le sous-réseau par défaut si aucune valeur n'est spécifiée.

- VpcId

  Type: String (chaîne)

  Par défaut : Valeur par défaut

  Description : (Obligatoire) Une nouvelle instance sera déployée dans cet Amazon Virtual Private Cloud (Amazon VPC) ou dans la valeur par défaut Amazon VPC si elle n'est pas spécifiée.

AWS-CreateManagedWindowsInstance

Description

Créez une instance EC2 pour Windows Server qui est configurée pour Systems Manager.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows

Paramètres

Parameters

- AmiId

  Type: String (chaîne)

  Par défaut : {{ssm:/aws/service/ami-windows-latest/Windows_Server-2016-English-Full-Base}}

  Description : (Obligatoire) ID d'AMI à utiliser pour lancer l'instance.
- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- GroupName

  Type: String (chaîne)

  Par défaut : SSMSecurityGroupForLinuxInstances

  Description : (Obligatoire) Nom du groupe de sécurité à créer.
- InstanceType

  Type: String (chaîne)

  Par défaut : t2.medium

  Description : (Obligatoire) Type d'instance à lancer. La valeur par défaut est t2.medium.
- KeyPairName

  Type: String (chaîne)

  Description : (Obligatoire) Paire de clés à utiliser lors de la création de l'instance.
- RemoteAccessCidr

  Type: String (chaîne)

  Par défaut : 0.0.0.0/0

  Description : (Obligatoire) Crée un groupe de sécurité avec le port RDP (plage de ports 3389) ouvert pour IPs être spécifié par CIDR (la valeur par défaut est 0.0.0.0/0). Si le groupe de sécurité existe déjà, il ne sera pas modifié et les règles ne changeront pas.
- RoleName

  Type: String (chaîne)

Par défaut : SSMManagedInstanceProfileRole

Description : (Obligatoire) Nom du rôle à créer.

- StackName

  Type: String (chaîne)

  Par défaut : CreateManagedInstanceStack{{automation:EXECUTION_ID}}

  Description : (Facultatif) Spécifiez le nom de la pile utilisé par ce document
- SubnetId

  Type: String (chaîne)

  Par défaut : Valeur par défaut

  Description : (Obligatoire) Une nouvelle instance sera déployée dans ce sous-réseau ou dans le sous-réseau par défaut si aucune valeur n'est spécifiée.
- VpcId

  Type: String (chaîne)

  Par défaut : Valeur par défaut

  Description : (Obligatoire) Une nouvelle instance sera déployée dans cet Amazon Virtual Private Cloud (Amazon VPC) ou dans la valeur par défaut Amazon VPC si elle n'est pas spécifiée.

AWS-CreateRdsSnapshot

Description

Créez un instantané Amazon Relational Database Service (Amazon RDS) pour une instance Amazon RDS.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Bases de données

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- DBInstanceIdentifier

  Type: String (chaîne)

Description : (Obligatoire) DBInstanceId ID de l'instance RDS à partir de laquelle créer un instantané.

- DBSnapshotIdentifier

  Type: String (chaîne)

  Description : (Facultatif) ID de l'instantané RDS à créer.DBSnapshotIdentifier
- InstanceTags

  Type: String (chaîne)

  Description : (Facultatif) Balises à créer pour l'instance.
- SnapshotTags

  Type: String (chaîne)

  Description : (Facultatif) Balises à créer pour l'instantané.

Étapes de document

createRDSSnapshot–Crée l'instantané RDS et renvoie l'ID de l'instantané.

verifyRDSSnapshot–Vérifie que l'instantané créé à l'étape précédente existe.

Résultats

createRDSSnapshot.SnapshotId – ID de l'instantané créé.

AWS-CreateServiceNowIncident

Description

Ce document crée un incident dans la table ServiceNow des incidents.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- Category

  Type: String (chaîne)

Description : (Facultatif) Catégorie de l'incident.

Valeurs valides : Aucune | Demande/Aide | Logiciel | Matériel | Réseau | Base de données

Valeur par défaut : None (Aucun)

- Description

  Type: String (chaîne)

  Description : (Obligatoire) Une explication détaillée de l'incident.
- Impact

  Type: String (chaîne)

  Description : (Facultatif) L'effet d'un incident sur les activités.

  Valeurs valides : Élevé | Moyen | Faible

  Valeur par défaut : Faible
- ServiceNowInstanceUsername

  Type: String (chaîne)

  Description : (Obligatoire) Nom de l'utilisateur avec lequel l'incident sera créé.
- ServiceNowInstancePassword

  Type: String (chaîne)

  Description : (Obligatoire) Nom d'un paramètre SSM chiffré contenant le mot de passe de l'ServiceNowutilisateur.
- ServiceNowInstanceURL

  Type: String (chaîne)

  Description : (Obligatoire) URL de l'ServiceNowinstance
- ShortDescription

  Type: String (chaîne)

  Description : (Obligatoire) Une brève description de l'incident.
- Sous-catégorie

  Type: String (chaîne)

  Description : (Facultatif) La sous-catégorie de l'incident.

  Valeurs valides : Aucun | Antivirus | E-mail | Application interne | Système d'exploitation | Système d'exploitation | UC | Disque | Clavier | Matériel | Mémoire | Surveillance | Souris | DHCP | DNS | Adresse IP | VPN | Sans fil | DB2 | MS SQL Server | Oracle

  Valeur par défaut : None (Aucun)

Étapes de document

Push_incident – – Pousse les informations sur l'incident vers ServiceNow .

Résultats

Push_incident.incidentID – ID d'incident créé.

AWS-CreateSnapshot

Description

Créez un instantané d'un volume Amazon EBS.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- Description

  Type: String (chaîne)

  Description : (Facultatif) Une description de l'instantané
- VolumeId

  Type: String (chaîne)

  Description : (Obligatoire) ID du volume.

AWS-DeleteCloudFormationStack

Description

Supprimez une pile AWS CloudFormation.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- StackNameOrId

  Type: String (chaîne)

  Description : (Obligatoire) Nom ou ID unique de la CloudFormation pile à supprimer

AWS-DeleteDynamoDbBackup

Description

Supprimez la sauvegarde d'une table Amazon DynamoDB.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- BackupArn

  Type: String (chaîne)

  Description : (Obligatoire) ARN de la sauvegarde de DynamoDB table à supprimer.

Sauvegardes AWS-DeleteDynamoDbTableBackups

Description

Supprimez les sauvegardes de table DynamoDB en fonction d'un nombre de jours de conservation ou d'un nombre de sauvegardes.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- LambdaAssumeRole

  Type: String (chaîne)

  Description : (Facultatif) ARN du rôle qui autorise Lambda créé par Automation à effectuer les actions en votre nom. Si vous ne spécifiez pas cette valeur, un rôle transitoire est créé pour exécuter la fonction Lambda.
- RetentionCount

  Type: String (chaîne)

  Par défaut : 10

  Description : (Facultatif) Nombre de sauvegardes à conserver pour la table. S'il existe plus que le nombre spécifié de sauvegardes, les sauvegardes les plus anciennes au-delà de ce nombre sont supprimées. ou RetentionCount peut être utilisé, pas les deux.RetentionDays
- RetentionDays

  Type: String (chaîne)

  Description : (Facultatif) Nombre de jours de conservation des sauvegardes de la table. Les sauvegardes plus ancienne que le nombre de jours spécifié sont supprimées. ou RetentionCount peut être utilisé, pas les deux.RetentionDays
- TableName

  Type: String (chaîne)

  Description : (Obligatoire) Nom de la DynamoDB table.

InstantanésVolumeSnapshots-DeleteEbs

Description

Supprimez un instantané d'un volume Amazon Elastic Block Store (Amazon EBS).

> **Note**
>
> La fonction AWS Lambda qui s'exécute pendant cette opération a un temps d'exécution maximal (délai d'expiration) de 60 secondes. Si vous avez un grand nombre d'instantanés de volume Amazon EBS à supprimer, l'opération peut échouer avec un message d'erreur.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- LambdaAssumeRole

  Type: String (chaîne)

  Description : (Facultatif) ARN du rôle qui autorise Lambda créé par Automation à effectuer les actions en votre nom. Si vous ne spécifiez pas cette valeur, un rôle transitoire est créé pour exécuter la fonction Lambda.
- RetentionCount

  Type: String (chaîne)

  Par défaut : 10

  Description : (Facultatif) Nombre d'instantanés à conserver pour le volume. Ou RetentionCount doit être mentionné, pas les deux.RetentionDays
- RetentionDays

  Type: String (chaîne)

  Description : (Facultatif) Nombre de jours de conservation des instantanés du volume. Soit RetentionCount doit être mentionné, pas les deuxRetentionDays
- VolumeId

  Type: String (chaîne)

  Description : (Obligatoire) Identificateur du volume pour lequel supprimer les instantanés.

AWS-DeleteImage

Description

Supprimez une Amazon Machine Image (AMI) et tous les instantanés associés.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- ImageId

  Type: String (chaîne)

  Description : (Obligatoire) ID de l'AMI.

Autorisations IAM nécessaires

`AutomationAssumeRole` nécessite les actions suivantes pour exécuter correctement le document Automation.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2:DeleteSnapshot",
            "Resource": "arn:aws:ec2:{region}::snapshot/*"
        },
        {
            "Effect": "Allow",
            "Action": "ec2:DescribeImages",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "ec2:DeregisterImage",
            "Resource": "*"
        }
    ]
}
```

AWSConfigRemediation-DeleteUnusedIAMGroup

Description

Le document d'automatisation AWSConfigRemediation -DeleteUnusedIAMGroup supprime un IAM groupe qui ne contient aucun IAM utilisateur.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- GroupName

  Type: String (chaîne)

  Description : (Obligatoire) Nom du IAM groupe que vous souhaitez supprimer.

Autorisations IAM nécessaires

`AutomationAssumeRole` nécessite les actions suivantes pour exécuter correctement le document Automation.

- `ssm:ExecuteAutomation`
- `ssm:GetAutomationExecution`
- `iam:DeleteGroup`
- `iam:DeleteGroupPolicy`
- `iam:DetachGroupPolicy`

Étapes de document

- aws:executeScript : supprime les IAM stratégies gérées et en ligne attachées au IAM groupe cible, puis supprime le IAM groupe.

AWS-DeleteSnapshot

Description

Supprimez un instantané d'un volume Amazon EBS.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- SnapshotId

  Type: String (chaîne)

  Description : (Obligatoire) ID de l'instantané EBS.

AWS-DetachEBSVolume

Description

Détachez un volume Amazon EBS d'une instance EC2.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- LambdaAssumeRole

  Type: String (chaîne)

  Description : (Facultatif) ARN du rôle assumé parLambda
- VolumeId

  Type: String (chaîne)

  Description : (Obligatoire) ID du volume EBS. Le volume et l'instance doivent être dans la même zone de disponibilité.

AWS-DisablePublicAccessForSecurityGroup

Description

Ce document désactive les ports SSH et RDP par défaut qui sont ouverts à toutes les adresses IP.

> **Important**
>
> Ce document échoue avec une erreur « InvalidPermission.NotFound » pour les groupes de sécurité qui répondent aux deux critères suivants : 1) Le groupe de sécurité se trouve dans un VPC autre que celui par défaut ; et 2) Les règles entrantes pour le groupe de sécurité ne spécifient pas de ports ouverts à l'aide des quatre modèles suivants :
>
> - `0.0.0.0/0`
> - `::/0`
> - `SSH or RDP port + 0.0.0.0/0`
> - `SSH or RDP port + ::/0`
>
> Si le groupe de sécurité se trouve dans un VPC autre que celui par défaut et, par exemple, spécifie des ports ouverts en utilisant uniquement le format `SSH or RDP port + 0.0.0.0/0`, l'exécution du document échoue.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- GroupId

  Type: String (chaîne)

  Description : (Obligatoire) ID du groupe de sécurité pour lequel les ports doivent être désactivés.
- IpAddressToBlock

  Type: String (chaîne)

  Description : (Facultatif) Adresses supplémentaires à partir desquelles l'accès doit être bloqué, au format IPv4 `.1.2.3.4/32`

Résultats

None (Aucun)

AWS-DisableS3BucketPublicReadWrite

Description

Utilisez Amazon Simple Storage Service (Amazon S3) `Block Public Access` pour désactiver l'accès en lecture et en écriture pour un compartiment S3 public. Pour plus d'informations, consultez Utilisation du blocage d'accès public Amazon S3, dans le Amazon Simple Storage Service Manuel du développeur.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- S3BucketName

  Type: String (chaîne)

  Description : (Obligatoire) Compartiment S3 au niveau duquel vous souhaitez restreindre l'accès.

AWS-EnableCloudTrail

Description

Créez un suivi AWS CloudTrail et configurer la journalisation d'un compartiment S3.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- S3BucketName

  Type: String (chaîne)

  Description : (Obligatoire) Nom du compartiment S3 désigné pour publier les fichiers journaux.

  > **Note**
  >
  > Le compartiment S3 doit exister et la stratégie de compartiment doit accorder à CloudTrail l'autorisation d'écrire dessus. Pour plus d'informations, consultez Stratégie de compartiment Amazon S3 pour CloudTrail .
- TrailName

  Type: String (chaîne)

  Description : (Obligatoire) Nom du nouveau journal de suivi.

AWSConfigRemediation-EnableEnhancedMonitoringOnRDSInstance

Description

Le document AWSConfigRemediation d'automatisation -EnableEnhancedMonitoringOnRDSInstance active la surveillance améliorée sur l'instance Amazon RDS de base de données que vous spécifiez. Pour plus d'informations sur la surveillance améliorée, consultez Surveillance améliorée dans le Amazon RDS Guide de l'utilisateur .

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Bases de données

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- MonitoringInterval

Type: Integer

Valeurs valides : 1 | 5 | 10 | 15 | 30 | 60

Description : (Obligatoire) Intervalle en secondes au cours duquel les métriques de surveillance améliorée sont collectées à partir de l'instance de base de données.

- MonitoringRoleArn

  Type: String (chaîne)

  Description : (Obligatoire) Amazon Resource Name (ARN) du IAM rôle qui autorise Amazon RDS à envoyer des métriques de surveillance améliorée à Amazon CloudWatch Logs .

- ResourceId

  Type: String (chaîne)

  Description : (Obligatoire) Identificateur de ressource de l'instance de base de données sur laquelle vous souhaitez activer la surveillance améliorée.

Autorisations IAM nécessaires

`AutomationAssumeRole` nécessite les actions suivantes pour exécuter correctement le document Automation.

- `ssm:ExecuteAutomation`
- `ssm:GetAutomationExecution`
- `rds:DescribeDBInstances`
- `rds:ModifyDBInstance`

Étapes de document

- aws:executeAwsApi - Recueille l'identifiant d'instance de base de données à partir de l'identifiant de ressource d'instance de base de données.
- aws:assertAwsResourceProperty : confirme que l'instance de base de données est à `AVAILABLE` l'état .
- aws:executeAwsApi - Active la surveillance améliorée sur votre instance de base de données.
- aws:executeScript - Confirme que la surveillance améliorée est activée sur votre instance de base de données.

AWS-EnableS3BucketEncryption

Description

Activez le chiffrement pour un compartiment Amazon Simple Storage Service (Amazon S3) (chiffrement du contenu du compartiment).

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- BucketName

  Type: String (chaîne)

  Description : (Obligatoire) Nom du compartiment S3 dans lequel vous souhaitez chiffrer le contenu.
- SSEAlgorithm

  Type: String (chaîne)

  Par défaut : AES256

  Description : (Facultatif) Algorithme de chiffrement côté serveur à utiliser pour le chiffrement par défaut.

AWSSupport-ExécuterEC2Rescue

Description

Ce document utilise l'EC2Rescueoutil pour dépanner et réparer, le cas échéant, les problèmes de connectivité courants avec l'instance EC2 spécifiée pour Linux ou Windows Server .

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- EC2RescueInstanceType

  Type: String (chaîne)

Valeurs valides : t2.small | t2.medium | t2.large

Par défaut : t2.small

Description : (Obligatoire) Type d'instance EC2 de l'EC2Rescueinstance. Taille recommandée : t2.small.

- LogDestination

  Type: String (chaîne)

  Description : (Facultatif) Nom du compartiment S3 de votre compte dans lequel vous souhaitez charger les journaux de dépannage. Assurez-vous que la stratégie de compartiment n'accorde pas des autorisations en lecture/écriture superflues pour les tiers qui n'ont pas besoin d'accéder aux journaux collectés.
- SubnetId

  Type: String (chaîne)

  Par défaut : CreateNewVPC

  Description : (Facultatif) ID de sous-réseau de l'EC2Rescueinstance. Par défaut, AWS Systems Manager Automation crée un VPC. Utilisez également SelectedInstanceSubnet pour utiliser le même sous-réseau que votre instance ou spécifiez un ID de sous-réseau personnalisé. IMPORTANT : Le sous-réseau doit être dans la même zone de disponibilité qu' UnreachableInstanceId et doit autoriser l'accès aux points de terminaison SSM.
- UnreachableInstanceId

  Type: String (chaîne)

  Description : (Obligatoire) ID de votre instance EC2 inaccessible. IMPORTANT :AWS Systems Manager Automation arrête cette instance et crée une AMI avant d'essayer toute autre opération. Les données stockées sur les volumes de stockage d'instance seront perdues. L'adresse IP publique change si vous n'utilisez pas une adresse IP Elastic.

Autorisations IAM nécessaires

`AutomationAssumeRole` nécessite les actions suivantes pour exécuter correctement le document Automation.

Vous devez avoir au moins ssm:ExecuteAutomation et ssm:GetAutomationExecution pour être en mesure de lire la sortie de l'automatisation. Pour plus d'informations sur les autorisations requises, consultez AWSSupport -StartEC2RescueWorkflow

Étapes de document

1. aws:assertAwsResourceProperty : vérifie que la plateforme de l'instance fournie est Windows.
   a. (EC2Rescue pour Windows) Si l'instance fournie est une instance Windows Server :
      i. aws:executeAutomation - Appeler AWSSupport -StartEC2RescueWorkflow avec le script hors connexion EC2Rescue pour Windows
      ii. aws:executeAwsApi : récupérez l'ID d'AMI de sauvegarde à partir de l'automatisation imbriquée
      iii. aws:executeAwsApi : récupérez le EC2Rescue résumé de l'automatisation imbriquée.
   b. (EC2Rescue pour Linux) Si l'instance fournie est une instance Linux :
      i. aws:executeAutomation - Appeler AWSSupport -StartEC2RescueWorkflow avec le script hors connexion EC2Rescue pour Linux
      ii. aws:executeAwsApi : récupérez l'ID d'AMI de sauvegarde à partir de l'automatisation imbriquée
      iii. aws:executeAwsApi : récupérez le EC2Rescue résumé de l'automatisation imbriquée.

Résultats

SortiegetEC2RescueForWindowsResult

getWindowsBackupAmiID d'image

SortiegetEC2RescueForLinuxResult

getLinuxBackupAmiID d'image

AWS-ExportOpsDataToS3

Description

Ce document récupère une liste de OpsData résumés dans AWS Systems Manager l'Explorateur et les exporte vers un objet dans un compartiment S3 spécifié.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- columnFields

  Type: StringList

  Description : (Obligatoire) Champs de colonne à écrire dans le fichier de sortie.
- filtres

  Type: String (chaîne)

  Description : (Facultatif) Filtres pour la getOpsSummary demande.
- resultAttribute

  Type: String (chaîne)

  Description : (Facultatif) Attribut de résultat pour getOpsSummary la demande.
- s3BucketName

  Type: String (chaîne)

  Description : (Obligatoire) Compartiment S3 dans lequel vous souhaitez télécharger le fichier de sortie.

- snsSuccessMessage

  Type: String (chaîne)

  Description : (Facultatif) Message à envoyer à la fin du document.
- snsTopicArn

  Type: String (chaîne)

  Description : (Obligatoire) ARN de la rubrique Amazon Simple Notification Service (Amazon SNS) à avertir lorsque le téléchargement est terminé.
- syncName

  Type: String (chaîne)

  Description : (Facultatif) Nom de la synchronisation des données de la ressource.

Étapes de document

getOpsSummaryStep–Récupère jusqu'à 5 000 résumés d'opérations à exporter dans un fichier CSV maintenant.

Résultats

OpsDataobject – Si le document Automation s'exécute correctement, vous trouverez l'OpsDataobjet exporté dans votre compartiment S3 cible.

AWSSupport-AccorderPermissionsToIAMUser

Description

Ce document accorde les autorisations spécifiées à un groupe IAM (nouveau ou existant) et ajoute l'utilisateur IAM existant. Stratégies que vous pouvez choisir : Facturation ou support. Pour activer l'accès à la facturation pour IAM, n'oubliez pas d'activer également l'accès des utilisateurs IAM et fédérés aux pages Billing and Cost Management.

> **Important**
>
> Si vous fournissez un groupe IAM, tous les utilisateurs IAM du groupe reçoivent les nouvelles autorisations.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

Type : chaîne

Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.

- IAMGroupName

  Type: String (chaîne)

  Par défaut : ExampleSupportAndBillingGroup

  Description : (Obligatoire) Il peut s'agir d'un groupe nouveau ou existant. Doit se conformer aux Limites des noms d'entité IAM.

- IAMUserName

  Type: String (chaîne)

  Par défaut : ExampleUser

  Description : (Obligatoire) Doit être un utilisateur existant.

- LambdaAssumeRole

  Type: String (chaîne)

  Description : (Facultatif) ARN du rôle assumé par lambda.

- Permissions

  Type: String (chaîne)

  Valeurs valides : SupportFullAccess| | |BillingFullAccessSupportAndBillingFullAccess

  Par défaut : SupportAndBillingFullAccess

  Description : (Obligatoire) Choisissez l'une des options suivantes : `SupportFullAccess`accorde un accès complet au Centre de support. `BillingFullAccess`accorde un accès complet au tableau de bord de facturation. `SupportAndBillingFullAccess`accorde un accès complet au centre de support et au tableau de bord de facturation. Plus d'informations sur les stratégies sous Détails du document.

Autorisations IAM nécessaires

`AutomationAssumeRole` nécessite les actions suivantes pour exécuter correctement le document Automation.

Les autorisations requises dépendent de la façon dont AWSSupport -GrantPermissionsToIAMUser est exécuté.

En cours d'exécution en tant qu'utilisateur ou rôle actuellement connecté

Il est recommandé que la stratégie gérée Amazon **AmazonSSMAutomationRole** soit attachée et que vous disposiez des autorisations supplémentaires suivantes pour pouvoir créer la fonction Lambda et le rôle IAM à transmettre à Lambda :

```
{
                    "Version": "2012-10-17",
                    "Statement": [
                        {
                            "Action": [
                                "lambda:InvokeFunction",
```

```
                                "lambda:CreateFunction",
                                "lambda:DeleteFunction",
                                "lambda:GetFunction"
                            ],
                            "Resource": "arn:aws:lambda:*:ACCOUNTID:function:AWSSupport-*",
                            "Effect": "Allow"
                        },
                        {
                            "Effect" : "Allow",
                            "Action" : [
                                "iam:CreateGroup",
                                "iam:AddUserToGroup",
                                "iam:ListAttachedGroupPolicies",
                                "iam:GetGroup",
                                "iam:GetUser"
                            ],
                            "Resource" : [
                                "arn:aws:iam::*:user/*",
                                "arn:aws:iam::*:group/*"
                            ]
                        },
                        {
                            "Effect" : "Allow",
                            "Action" : [
                                "iam:AttachGroupPolicy"
                            ],
                            "Resource": "*",
                            "Condition": {
                                "ArnEquals": {
                                    "iam:PolicyArn": [
                                        "arn:aws:iam::aws:policy/job-function/Billing",
                                        "arn:aws:iam::aws:policy/AWSSupportAccess"
                                    ]
                                }
                            }
                        },
                        {
                            "Effect" : "Allow",
                            "Action" : [
                                "iam:ListAccountAliases",
                                "iam:GetAccountSummary"
                            ],
                            "Resource" : "*"
                        }
                    ]
                }
```

Utilisation d' AutomationAssumeRole et d' LambdaAssumeRole

L'utilisateur doit disposer des autorisations ssm:ExecuteAutomation sur le document et iam:PassRole sur les rôles IAM transmis comme AutomationAssumeRole et LambdaAssumeRole . Voici les autorisations dont chaque rôle IAM a besoin :

```
AutomationAssumeRole

                    {
                        "Version": "2012-10-17",
                        "Statement": [
                            {
                                "Action": [
                                    "lambda:InvokeFunction",
                                    "lambda:CreateFunction",
                                    "lambda:DeleteFunction",
                                    "lambda:GetFunction"
```

```
                                ],
                                "Resource":
 "arn:aws:lambda:*:ACCOUNTID:function:AWSSupport-*",
                                "Effect": "Allow"
                            }
                        ]
                    }
```

```
LambdaAssumeRole

                {
                    "Version": "2012-10-17",
                    "Statement": [
                        {
                            "Effect" : "Allow",
                            "Action" : [
                                "iam:CreateGroup",
                                "iam:AddUserToGroup",
                                "iam:ListAttachedGroupPolicies",
                                "iam:GetGroup",
                                "iam:GetUser"
                            ],
                            "Resource" : [
                                "arn:aws:iam::*:user/*",
                                "arn:aws:iam::*:group/*"
                            ]
                        },
                        {
                            "Effect" : "Allow",
                            "Action" : [
                                "iam:AttachGroupPolicy"
                            ],
                            "Resource": "*",
                            "Condition": {
                                "ArnEquals": {
                                    "iam:PolicyArn": [
                                        "arn:aws:iam::aws:policy/job-function/Billing",
                                        "arn:aws:iam::aws:policy/AWSSupportAccess"
                                    ]
                                }
                            }
                        },
                        {
                            "Effect" : "Allow",
                            "Action" : [
                                "iam:ListAccountAliases",
                                "iam:GetAccountSummary"
                            ],
                            "Resource" : "*"
                        }
                    ]
                }
```

Étapes de document

1. aws:createStack : Exécutez un modèle AWS CloudFormation pour créer une fonctionnalité Lambda.
2. aws:invokeLambdaFunction : Exécutez Lambda pour définir des autorisations IAM.
3. aws:deleteStack : CloudFormation supprimez le modèle.

Résultats

configureIAM.Charge utile

AWSSupport-ListEC2Ressources

Description

Le document AWSSupport -ListEC2Resources Automation renvoie des informations sur Amazon EC2 les instances et les ressources connexes telles que les volumes Amazon Elastic Block Store (Amazon EBS), les adresses IP Elastic et les Amazon EC2 Auto Scaling groupes des AWS régions que vous spécifiez. Par défaut, les informations sont collectées à partir de toutes les régions et affichées dans la sortie de l'automatisation. Le cas échéant, vous pouvez spécifier un compartiment Amazon Simple Storage Service (Amazon S3) pour les informations à charger dans sous la forme d'un fichier de valeurs séparées par des virgules (.csv).

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- Bucket

  Type: String (chaîne)

  Description : (Facultatif) Nom du compartiment S3 dans lequel les informations collectées sont chargées.
- DisplayResourceDeletionDocumentation

  Type: String (chaîne)

  Par défaut : True

  Description : (Facultatif) Si cette option est définie sur `True` , l'automatisation crée des liens dans la sortie vers la documentation relative à la suppression de vos ressources.
- RegionsToQuery

  Type: String (chaîne)

  Par défaut : Toutes

  Description : (Facultatif) Les régions à partir desquelles vous souhaitez collecter les informations Amazon EC2 connexes.

Autorisations IAM nécessaires

`AutomationAssumeRole` nécessite les actions suivantes pour exécuter correctement le document Automation.

- `autoscaling:DescribeAutoScalingGroups`
- `ec2:DescribeAddresses`
- `ec2:DescribeImages`
- `ec2:DescribeInstances`
- `ec2:DescribeNetworkInterfaces`
- `ec2:DescribeRegions`
- `ec2:DescribeVolumes`
- `ec2:DescribeSnapshots`
- `elasticloadbalancing:DescribeLoadBalancers`

En outre, pour charger correctement les informations recueillies dans le compartiment S3 que vous spécifiez, l' `AutomationAssumeRole` nécessite les actions suivantes :

- `s3:GetBucketAcl`
- `s3:GetBucketPolicyStatus`
- `s3:PutObject`

Étapes de document

- aws:executeAwsApi - Collecte les régions activées pour le compte.
- aws:executeScript : confirme que les régions activées pour le compte prennent en charge les régions spécifiées dans le `RegionsToQuery` paramètre.
- aws:branch - Si aucune région n'est activée pour le compte, l'automatisation se termine.
- aws:executeScript - Répertorie toutes les instances EC2 pour le compte et les régions que vous spécifiez.
- aws:executeScript : répertorie toutes les Amazon Machine Images (AMI) pour le compte et les régions que vous spécifiez.
- aws:executeScript - Répertorie tous les volumes EBS pour le compte et les régions que vous spécifiez.
- aws:executeScript - Répertorie toutes les adresses IP Elastic pour le compte et les régions que vous spécifiez.
- aws:executeScript - Répertorie toutes les interfaces réseau Elastic pour le compte et les régions que vous spécifiez.
- aws:executeScript - Répertorie tous les Auto Scaling groupes pour le compte et les régions que vous spécifiez.
- aws:executeScript - Répertorie tous les équilibreurs de charge pour le compte et les régions que vous spécifiez.
- aws:executeScript : charge les informations collectées dans le compartiment S3 spécifié si vous fournissez une valeur pour le `Bucket` paramètre.

AWSSupport-ManageRDPSettings

Description

Le document d'Automation AWSSupport -ManageRDPSettings permet à l'utilisateur de gérer les paramètres RDP (Remote Desktop Protocol) courants, tels que le port RDP et l'authentification NLA (Network Layer Authentication). Par défaut, le document lit et produit les valeurs de ces paramètres.

> **Important**
>
> Les modifications apportées aux paramètres RDP doivent être vérifiées minutieusement avant d'exécuter ce document.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- InstanceId

  Type: String (chaîne)

  Description : (Obligatoire) ID de l'instance gérée dont les paramètres RDP doivent être gérés.
- NLASettingAction

  Type: String (chaîne)

  Valeurs valides : Check | Enable | Disable (Vérifier | Activer | Désactiver)

  Par défaut : Check

  Description : (Obligatoire) Une action à effectuer sur le paramètre NLA : Cochez, activez, désactivez.
- RDPPort

  Type: String (chaîne)

  Par défaut : 3389

  Description : (Facultatif) Spécifiez le nouveau port RDP. Utilisé uniquement lorsque l'action est définie sur Modify. Le numéro de port doit être compris entre 1025 et 65535. Remarque: Une fois le port modifié, le service RDP est redémarré.
- RDPPortAction

  Type: String (chaîne)

  Valeurs valides : Vérifier | Modifier

  Par défaut : Check

  Description : (Obligatoire) Action à appliquer au port RDP.

- RemoteConnections

  Type: String (chaîne)

  Valeurs valides : Check | Enable | Disable (Vérifier | Activer | Désactiver)

  Par défaut : Check

  Description : (Obligatoire) Action à effectuer au niveau du fDenyTSConnections paramètre.

Autorisations IAM nécessaires

`AutomationAssumeRole` nécessite les actions suivantes pour exécuter correctement le document Automation.

L'instance EC2 recevant la commande doit avoir un rôle IAM avec la stratégie gérée AmazonSSMManagedInstanceCore Amazon attachée. L'utilisateur doit avoir au moins ssm:SendCommand pour envoyer la commande à l'instance, plus ssm:GetCommandInvocation pour pouvoir lire la sortie de la commande.

Étapes de document

aws:runCommand : Exécutez le PowerShell script pour modifier ou vérifier les paramètres RDP sur l'instance cible.

Résultats

.SortiemanageRDPSettings

AWSSupport-ManageWindowsService

Description

Le document d'automatisation AWSSupport -ManageWindowsService permet à un utilisateur d'arrêter, de démarrer, de redémarrer, de suspendre ou de désactiver n'importe quel service Windows sur l'instance cible.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.

- InstanceId

  Type: String (chaîne)

  Description : (Obligatoire) ID de l'instance gérée dont les paramètres RDP doivent être gérés.
- ServiceAction

  Type: String (chaîne)

  Valeurs valides : Check | Restart | Force-Restart | Start | Stop | Force-Stop | Pause

  Par défaut : Check

  Description : (Obligatoire) Action à appliquer au service Windows. Notez que `Force-Restart` et `Force-Stop` peuvent être utilisés pour redémarrer et arrêter un service qui possède des services dépendants.
- StartupType

  Type: String (chaîne)

  Valeurs valides : Vérifier | Auto | Demande | Désactivé |DelayedAutoStart

  Par défaut : Check

  Description : (Obligatoire) Type de démarrage à appliquer au service Windows.
- WindowsServiceName

  Type: String (chaîne)

  Description : (Obligatoire) Nom de service Windows valide.

Autorisations IAM nécessaires

`AutomationAssumeRole` nécessite les actions suivantes pour exécuter correctement le document Automation.

Il est recommandé que l'instance EC2 recevant la commande dispose d'un rôle IAM avec la stratégie gérée AmazonSSMManagedInstanceCore Amazon attachée. L'utilisateur doit avoir au moins ssm:ExecuteAutomation et ssm:SendCommand pour exécuter Automation et envoyer la commande à l'instance, plus ssm:GetAutomationExecution pour pouvoir lire la sortie d'Automation.

Étapes de document

aws:runCommand : Exécutez le PowerShell script pour appliquer la configuration souhaitée au service Windows sur l'instance cible.

Résultats

.SortiemanageWindowsService

Instance_correctif_AWS

Description

Appliquez les correctifs aux instances EC2 d'un groupe Auto Scaling.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- InstanceId

  Type: String (chaîne)

  Description : (Obligatoire) ID de l'instance à corriger. Ne spécifiez pas un ID d'instance qui est configuré pour s'exécuter au cours d'une fenêtre de maintenance.
- LambdaRoleArn

  Type: String (chaîne)

  Description : (Facultatif) ARN du rôle qui autorise Lambda créé par Automation à effectuer les actions en votre nom. Si vous ne spécifiez pas cette valeur, un rôle transitoire est créé pour exécuter la fonction Lambda.
- WaitForInstance

  Type: String (chaîne)

  Par défaut : PT2

  Description : (Facultatif) Durée pendant laquelle l'automatisation doit attendre pour permettre à l'instance de se remettre en service.
- WaitForReboot

  Type: String (chaîne)

  Par défaut : PT5

  Description : (Facultatif) Durée pendant laquelle l'automatisation doit attendre pour permettre à une instance corrigée de redémarrer.

AWS-PatchInstanceWithRollback

Description

Conforme l'instance EC2 au niveau de base ; restaure le volume racine en cas d'échec.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- InstanceId

  Type: String (chaîne)

  Description : (Obligatoire) EC2 InstanceId auquel nous appliquons le correctif de base.
- LambdaAssumeRole

  Type: String (chaîne)

  Description : (Facultatif) ARN du rôle qui autorise Lambda créé par Automation à effectuer les actions en votre nom. Si vous ne spécifiez pas cette valeur, un rôle transitoire est créé pour exécuter la fonction Lambda.
- ReportS3Bucket

  Type: String (chaîne)

  Description : (Facultatif) Destination Amazon S3 du compartiment pour le rapport de conformité généré au cours du processus.

Étapes de document

Numéro de l'étape	Nom de l'étape	Action Automation
1	createDocumentStack	aws:createStack
2	IdentifyRootVolume	aws:invokeLambdaFunction
3	PrePatchSnapshot	aws:executeAutomation
4	installMissingUpdates	aws:runCommand
5	SleepThruInstallation	aws:invokeLambdaFunction
6	CheckCompliance	aws:invokeLambdaFunction
7	SaveComplianceReportToS3	aws:invokeLambdaFunction
8	ReportSuccessOrFailure	aws:invokeLambdaFunction
9	DeleteSnapshot	aws:invokeLambdaFunction
10	createDocumentStack	aws:deleteStack

Résultats

IdentifyRootVolume.Charge utile

.SortiePrePatchSnapshot

SaveComplianceReportToS3.Charge utile

RestoreFromSnapshot.Charge utile

CheckCompliance.Charge utile

AWS-PublishSNSNotification

Description

Publiez une notification vers Amazon SNS.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- Message

  Type: String (chaîne)

  Description : (Obligatoire) Message à inclure dans la notification SNS.
- TopicArn

  Type: String (chaîne)

  Description : (Obligatoire) ARN de la rubrique SNS dans laquelle publier la notification.

AWS-RebootRdsInstance

Description

Le document AWS-RebootRdsInstance Automation redémarre une instance de base de données Amazon Relational Database Service (Amazon RDS) si elle n'a pas déjà redémarré.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Bases de données

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- InstanceId

  Type: String (chaîne)

  Description : (Obligatoire) ID de l'instance Amazon RDS de base de données que vous souhaitez redémarrer.

Étapes de document

RebootInstance- Redémarre l'instance de base de données si elle n'est pas déjà en cours de redémarrage.

WaitForAvailableState- Attend que l'instance de base de données termine le processus de redémarrage.

Résultats

L'exécution de l'automatisation n'a pas de sortie.

AWS-ReleaseElasticIP

Description

Libérer l'adresse IP Elastic spécifiée à l'aide de l'ID d'allocation

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- AllocationId

  Type: String (chaîne)

  Description : (Obligatoire) ID d'allocation de l'adresse IP Elastic.

AWSSupport-ResetAccess (RéinitialiserAccès)

Description

Ce document utilise l'EC2Rescueoutil sur l'instance EC2 spécifiée pour réactiver le déchiffrement de mot de passe via la console EC2 (Windows), ou pour générer et ajouter une nouvelle paire de clés SSH (Linux). Si vous avez oublié votre paire de clés, cette automatisation créera un mot de passe IAM que vous pourrez utiliser pour lancer une nouvelle instance EC2 avec une paire de clés qui vous appartient (Windows).

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- EC2RescueInstanceType

  Type: String (chaîne)

  Valeurs valides : t2.small | t2.medium | t2.large

  Par défaut : t2.small

  Description : (Obligatoire) Type d'instance EC2 de l'EC2Rescueinstance. Taille recommandée : t2.small.
- InstanceId

  Type: String (chaîne)

  Description : (Obligatoire) ID de l'instance EC2 pour laquelle vous souhaitez réinitialiser l'accès.

> **Important**
>
> Systems ManagerAutomation arrête cette instance et crée une AMI avant d'essayer toute autre opération. Les données stockées sur les volumes de stockage d'instance seront perdues. L'adresse IP publique change si vous n'utilisez pas une adresse IP Elastic.

- SubnetId

  Type: String (chaîne)

  Par défaut : CreateNewVPC

  Description : (Facultatif) ID de sous-réseau de l'EC2Rescueinstance. Par défaut, Systems Manager Automation crée un VPC. Sinon, utilisez SelectedInstanceSubnet pour utiliser le même sous-réseau que votre instance, ou spécifiez un ID de sous-réseau personnalisé.

  > **Important**
  >
  > Le sous-réseau doit être dans la même zone de disponibilité qu' InstanceId et doit autoriser l'accès aux points de terminaison SSM.

Autorisations IAM nécessaires

`AutomationAssumeRole` nécessite les actions suivantes pour exécuter correctement le document Automation.

Vous devez avoir au moins ssm:ExecuteAutomation, ssm:GetParameter (pour récupérer le nom de paramètre de la clé SSH) et ssm:GetAutomationExecution pour pouvoir lire la sortie de l'automatisation. Pour de plus amples informations sur les autorisations requises, veuillez consulter AWSSupport-StartEC2RescueWorkflow (p. 600).

Étapes de document

1. aws:assertAwsResourceProperty : vérifie que la plateforme de l'instance fournie est Windows.
   a. (EC2Rescue pour Windows) Si l'instance fournie est Windows :
      i. aws:executeAutomation - Appeler AWSSupport -StartEC2RescueWorkflow avec le script EC2Rescue de réinitialisation du mot de passe hors connexion pour Windows
      ii. aws:executeAwsApi : récupérez l'ID d'AMI de sauvegarde à partir de l'automatisation imbriquée
      iii. aws:executeAwsApi : récupérez l'ID d'AMI activé pour le mot de passe à partir de l'automatisation imbriquée
      iv. aws:executeAwsApi : récupérez le EC2Rescue résumé de l'automatisation imbriquée.
   b. (EC2Rescue pour Linux) Si l'instance fournie est Linux :
      i. aws:executeAutomation - Invoke AWSSupport -StartEC2RescueWorkflow avec le script d'injection de clé SSH EC2Rescue pour Linux hors connexion
      ii. aws:executeAwsApi : récupérez l'ID d'AMI de sauvegarde à partir de l'automatisation imbriquée
      iii. aws:executeAwsApi : récupérez le nom de paramètre SSM pour la clé SSH injectée
      iv. aws:executeAwsApi : récupérez le EC2Rescue résumé de l'automatisation imbriquée.

Résultats

.SortiegetEC2RescueForWindowsResult

getWindowsBackupAmiID d'image

getWindowsPasswordEnabledAmiID d'image

.SortiegetEC2RescueForLinuxResult

getLinuxBackupAmiID d'image

getLinuxSSHKeyParameter.Nom

AWS-ResizeInstance

Description

Modifiez le type d'une instance EC2.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- InstanceId

  Type: String (chaîne)

  Description : (Obligatoire) ID de l'instance.
- InstanceType

  Type: String (chaîne)

  Description : (Obligatoire) Type d'instance.
- LambdaAssumeRole

  Type: String (chaîne)

  Description : (Facultatif) ARN du rôle assumé par Lambda .

AWS-RestartEC2Instance

Description

Redémarrez une ou plusieurs instances EC2.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- InstanceId

  Type: StringList

  Description : (Obligatoire) Instance(s) EC2 à redémarrer

AWS-RunCfnLint

Description

Ce document utilise un Linter AWS CloudFormation (`cfn-python-lint`) pour valider les modèles YAML et JSON par rapport à la spécification de ressource AWS CloudFormation. Le document `AWS-RunCfnLint` effectue des vérifications supplémentaires. Par exemple, il s'assure que des valeurs valides ont été saisies pour les propriétés de ressource. Si la validation échoue, l'étape `RunCfnLintAgainstTemplate` échoue et la sortie de l'outil linter est fournie dans un message d'erreur. Ce document Automation utilise cfn-lint v0.24.4.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.

- ConfigureRuleFlag

  Type: String (chaîne)

  Description : (Facultatif) Options de configuration pour une règle à transmettre à la `--configure-rule` Paramètre .

  Exemple: E2001:strict=false, E3012:strict=false.
- FormatFlag

  Type: String (chaîne)

  Description : (Facultatif) Valeur à transmettre à l' `--format` pour spécifier le format de sortie.

  Valeurs valides : Par défaut | quiet | analysable | json

  Par défaut : Valeur par défaut
- IgnoreChecksFlag

  Type: String (chaîne)

  Description : (Facultatif) IDs des règles à transmettre au paramètre --ignore-checks. Ces règles ne sont pas vérifiées.

  Exemple: E1001, E100 à 3, W7001
- IncludeChecksFlag

  Type: String (chaîne)

  Description : (Facultatif) IDs des règles à transmettre à `--include-checks` Paramètre . Ces règles sont vérifiées.

  Exemple: E1001, E100 à 3, W7001
- InfoFlag

  Type: String (chaîne)

  Description : (Facultatif) Option pour l' `--info` Paramètre . Incluez l'option permettant d'activation des informations de journalisation supplémentaires sur le traitement du modèle.

  Par défaut : Faux
- TemplateFileName

  Type: String (chaîne)

  Description : Nom, ou clé, du fichier de modèle dans le compartiment S3.
- TemplateS3BucketName

  Type: String (chaîne)

  Description : Nom du compartiment S3 contenant le modèle de packer.
- RegionsFlag

  Type: String (chaîne)

  Description : (Facultatif) Valeurs à transmettre au pour `--regions` pour tester le modèle par rapport aux régions AWS spécifiées.

  Exemple : us-east-1, us-west-1

Étapes de document

RunCfnLintAgainstTemplate–Exécute l'`cfn-python-lint`outil par rapport au AWS CloudFormation modèle spécifié.

Résultats

RunCfnLintAgainstTemplate.output – Sortie stdout de l'`cfn-python-lint`outil.

AWS-RunPacker

Description

Ce document utilise l'HashiCorpoutil Packer pour valider, corriger ou créer des modèles de packer utilisés pour créer des images de machine. Ce document utilise Packer v1.4.4.

> **Note**
>
> Si vous spécifiez une valeur `vpc_id`, vous devez également spécifier la valeur `subnet_id` d'un sous-réseau public. À moins que vous ne modifiiez l'attribut d'adressage IPv4 public de votre sous-réseau, vous devez également définir sur true.`associate_public_ip_address`

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- Force

  Type: Booléen

  Description : Option Packer pour forcer l'exécution d'un générateur lorsque les artefacts d'une génération précédente empêchent l'exécution d'une génération.
- Mode

  Type: String (chaîne)

  Description : Mode, ou commande, d'utilisation de Packer lors de la validation par rapport au modèle. Les options incluent `Build` , et `Validate` .`Fix`
- TemplateFileName

  Type: String (chaîne)

Description : Nom, ou clé, du fichier de modèle dans le compartiment S3.

- TemplateS3BucketName

  Type: String (chaîne)

  Description : Nom du compartiment S3 contenant le modèle de packer.

Étapes de document

RunPackerProcessTemplate–Exécute le mode sélectionné par rapport au modèle à l'aide de l'outil Packer.

Résultats

RunPackerProcessTemplate.output – Sortie stdout de l'outil Packer.

RunPackerProcessTemplate.fixed_template_key Nom du modèle stocké dans un compartiment S3 à utiliser uniquement lors de l'exécution en mode « Fix ».–

RunPackerProcessTemplate.s3_bucket Nom du compartiment S3 qui contient le modèle fixe à utiliser uniquement lors de l'exécution en mode « Fix ».–

AWSSupport-Envoyer le compartiment LogBundleToS3

Description

Le document AWSSupport Automation -SendLogBundleToS3Bucket charge un groupe de journaux généré par l'EC2Rescueoutil à partir de l'instance cible dans le compartiment S3 spécifié. L'exécution d'Automation installe la version spécifique de EC2Rescue en fonction de la plateforme de l'instance cible. EC2Rescueest ensuite utilisé pour collecter tous les journaux de système d'exploitation disponibles.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- InstanceId

  Type: String (chaîne)

  Description : (Obligatoire) ID de l'instance gérée Windows ou Linux à partir de laquelle vous souhaitez collecter les journaux.

- S3BucketName

  Type: String (chaîne)

  Description : (Obligatoire) Compartiment S3 dans lequel charger les journaux.
- S3Path

  Type: String (chaîne)

  Par défaut : AWSSupport-EnvoyerLogBundleToS3Bucket/

  Description : (Facultatif) Chemin d'accès S3 pour les journaux collectés.

Autorisations IAM nécessaires

`AutomationAssumeRole` nécessite les actions suivantes pour exécuter correctement le document Automation.

Il est recommandé que l'instance EC2 recevant la commande dispose d'un rôle IAM avec la stratégie gérée AmazonSSMManagedInstanceCore Amazon attachée. L'utilisateur doit avoir au moins ssm:ExecuteAutomation et ssm:SendCommand pour exécuter Automation et envoyer la commande à l'instance, plus ssm:GetAutomationExecution pour pouvoir lire la sortie d'Automation.

Étapes de document

1. aws:runCommand : effectuez lEC2Rescue'installation via AWS-ConfigureAWSPackage.
2. aws:runCommand : Exécutez le PowerShell script pour collecter les journaux de dépannage Windows avec EC2Rescue .
3. aws:runCommand : Exécutez le script bash pour collecter les journaux de dépannage Linux avec EC2Rescue .

Résultats

.SortiecollectAndUploadWindowsLogBundle

.SortiecollectAndUploadLinuxLogBundle

AWS-SetupInventory

Description

Créez une association d'inventaire Systems Manager pour une ou plusieurs instances gérées. Le système recueille les métadonnées à partir de vos instances selon la planification dans l'association. Pour plus d'informations, consultez la section Inventaire AWS Systems Manager (p. 791).

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- Applications

  Type: String (chaîne)

  Par défaut : Enabled

  Description : (Facultatif) Collectez des métadonnées sur les applications installées.
- AssociatedDocName

  Type: String (chaîne)

  Par défaut : Inventaire des logiciels AWS-Gather

  Description : (Facultatif) Nom du document SSM utilisé pour collecter l'inventaire à partir de l'instance gérée.
- AssociationName

  Type: String (chaîne)

  Description : (Facultatif) Nom de l'association d'inventaire qui sera attribuée à l'instance.
- AssocWaitTime

  Type: String (chaîne)

  Par défaut : PT5

  Description : (Facultatif) Durée pendant laquelle la collecte d'inventaire doit s'interrompre lorsque l'heure de début de l'association d'inventaire est atteinte. L'heure utilise le format ISO 8601.
- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- AwsComponents

  Type: String (chaîne)

  Par défaut : Enabled

  Description : (Facultatif) Collectez les métadonnées des composants AWS tels qu'amazon-ssm-agent.
- CustomInventory

  Type: String (chaîne)

  Par défaut : Enabled

  Description : (Facultatif) Collectez les métadonnées d'inventaire personnalisé.
- Dépôt de

  Type: String (chaîne)

  Description : (Facultatif) Collectez des métadonnées sur les fichiers de vos instances. Pour de plus amples informations sur la façon de collecter ce type de données d'inventaire, veuillez consulter Utilisation de l'inventaire de fichiers et du registre Windows (p. 798). Nécessite désormais SSMAgent version 2.2.64.0 ou ultérieure. Exemple Linux : [{"Path":"/usr/bin", "Pattern":

["aws*", "*ssm*"],"Recursive":false},{"Path":"/var/log", "Pattern":["amazon*.*"], "Recursive":true, "DirScanLimit":1000}] Exemple Windows : [{"Path":"%PROGRAMFILES%", "Pattern": ["*.exe"],"Recursive":true}]

- InstanceDetailedInformation

  Type: String (chaîne)

  Par défaut : Enabled

  Description : (Facultatif) Collectez des informations supplémentaires sur l'instance, y compris le modèle d'UC, la vitesse et le nombre de cœurs, pour n'en citer que quelques-uns.
- InstanceIds

  Type: String (chaîne)

  Par défaut : *

  Description : (Obligatoire) Instances EC2 que vous souhaitez inventorier.
- LambdaAssumeRole

  Type: String (chaîne)

  Description : (Facultatif) ARN du rôle qui autorise Lambda créé par Automation à effectuer les actions en votre nom. Si vous ne spécifiez pas cette valeur, un rôle transitoire est créé pour exécuter la fonction Lambda.
- NetworkConfig

  Type: String (chaîne)

  Par défaut : Enabled

  Description : (Facultatif) Collectez des métadonnées sur les configurations réseau.
- OutputS3BucketName

  Type: String (chaîne)

  Description : (Facultatif) Nom d'un Amazon S3 compartiment dans lequel vous souhaitez écrire les données des journaux d'inventaire.
- OutputS3KeyPrefix

  Type: String (chaîne)

  Description : (Facultatif) Préfixe Amazon S3 de clé (sous-dossier) dans lequel vous souhaitez écrire les données des journaux d'inventaire.
- OutputS3Region

  Type: String (chaîne)

  Description : (Facultatif) Le nom de la AWS région dans laquelle le Amazon S3 existe.
- Schedule

  Type: String (chaîne)

  Par défaut : cron(0 */30 * * * ? *)

  Description : (Facultatif) Expression cron pour la planification de l'association d'inventaire. La valeur par défaut est toutes les 30 minutes.
- Services

Type: String (chaîne)

Par défaut : Enabled

Description : (Facultatif, système d'exploitation Windows uniquement, nécessite SSMAgent la version 2.2.64.0 et ultérieure) Collectez des données pour les configurations de service.

- WindowsRegistry

  Type: String (chaîne)

  Description : (Facultatif) Collectez les métadonnées relatives aux clés de registre Microsoft Windows. Pour de plus amples informations sur la façon de collecter ce type de données d'inventaire, veuillez consulter Utilisation de l'inventaire de fichiers et du registre Windows (p. 798). Nécessite désormais Agent SSM version 2.2.64.0 ou ultérieure. Exemple: [{"Path":"HKEY_CURRENT_CONFIG\\System","Recursive":true},{"Path":"HKEY_LOCAL_MACHINE\\SOFTWARE\\Amazon\\MachineImage", "ValueNames":["AMIName"]]

- WindowsRoles

  Type: String (chaîne)

  Par défaut : Enabled

  Description : (Facultatif) Collectez des informations sur les rôles Windows sur l'instance. S'applique aux systèmes d'exploitation Windows uniquement. Nécessite désormais SSMAgent version 2.2.64.0 ou ultérieure.

- WindowsUpdates

  Type: String (chaîne)

  Par défaut : Enabled

  Description : (Facultatif) Collectez des données sur toutes les mises à jour Windows de l'instance.

AWSSupport-SetupIPMonitoringFromVPC

Description

AWSSupport-SetupIPMonitoringFromVPC crée une instance EC2 dans le sous-réseau spécifié et surveille la cible sélectionnée IPs (IPv4 ou IPv6) en exécutant en continu des tests ping, MTR, traceroute et tracetcp. Les résultats sont stockés dans des journaux Amazon CloudWatch Logs, et des filtres de métriques sont appliqués pour visualiser rapidement la latence et les statistiques de perte de paquets dans un tableau de bord CloudWatch.

Informations supplémentaires

Les données CloudWatch Logs peuvent être utilisées pour le dépannage et l'analyse des modèles/tendances. De plus, vous pouvez configurer des alarmes CloudWatch avec des notifications Amazon SNS lorsque la perte de paquets et/ou la latence atteignent un seuil. Les données peuvent également être utilisées lors de l'ouverture d'une demande Premium Support, pour vous aider à isoler un problème rapidement et à réduire les temps de résolution lors de l'investigation d'un problème de réseau.

> **Note**
>
> Pour nettoyer les ressources créées par AWSSupport -SetupIPMonitoringFromVPC, vous pouvez exécuter le document Automation`AWSSupport-TerminateIPMonitoringFromVPC`. Pour plus d'informations, consultez AWSSupport-TerminateIPMonitoringDeVPC (p. 609) .

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- CloudWatchLogGroupNamePrefix

  Type: String (chaîne)

  Par défaut : /AWSSupport-SetupIPMonitoringDeVPC

  Description : (Facultatif) Préfixe utilisé pour chaque groupe de CloudWatch journaux créé pour les résultats du test.
- CloudWatchLogGroupRetentionInDays

  Type: String (chaîne)

  Valeurs valides : 1 | 3 | 5 | 7 | 14 | 30 | 60 | 90 | 120 | 150 | 180 | 365 | 400 | 545 | 731 | 1827 | 3653

  Par défaut : 7

  Description : (Facultatif) Nombre de jours pendant lesquels vous souhaitez conserver les résultats de la surveillance du réseau.
- InstanceType

  Type: String (chaîne)

  Valeurs valides : t2.micro | t2.small | t2.medium | t2.large

  Par défaut : t2.micro

  Description : (Facultatif) Type d'instance EC2 de l'EC2Rescueinstance. Taille recommandée : t2.micro.
- SubnetId

  Type: String (chaîne)

  Description : (Obligatoire) ID de sous-réseau pour l'instance de surveillance. Sachez que si vous spécifiez un sous-réseau privé, vous devez vous assurer qu'il existe un accès Internet pour permettre à l'instance de surveillance de configurer le test (c'est-à-dire, installer l'agent CloudWatch Logs, interagir avec Systems Manager et CloudWatch).
- TargetIPs

Type: String (chaîne)

Description : (Obligatoire) Liste séparée par des virgules de IPv4s et/ou IPv6s à surveiller. Les espaces ne sont pas autorisés. La taille maximale est de 255 caractères. Sachez que si vous fournissez une adresse IP non valide, l'automatisation échoue et restaure la configuration du test.

Autorisations IAM nécessaires

`AutomationAssumeRole` nécessite les actions suivantes pour exécuter correctement le document Automation.

Il est recommandé que la stratégie AmazonSSMAutomationRole IAM gérée soit attachée à l'utilisateur qui exécute l'automatisation. En outre, l'utilisateur doit disposer de la stratégie suivante associée à son compte utilisateur, groupe ou rôle :

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "iam:CreateRole",
                "iam:CreateInstanceProfile",
                "iam:GetRole",
                "iam:GetInstanceProfile",
                "iam:DetachRolePolicy",
                "iam:AttachRolePolicy",
                "iam:PassRole",
                "iam:AddRoleToInstanceProfile",
                "iam:RemoveRoleFromInstanceProfile",
                "iam:DeleteRole",
                "iam:DeleteInstanceProfile",
                "iam:PutRolePolicy",
                "iam:DeleteRolePolicy"
            ],
            "Resource": [
                "arn:aws:iam::AWS_account_ID:role/AWSSupport/SetupIPMonitoringFromVPC_*",
                "arn:aws:iam::AWS_account_ID:instance-profile/AWSSupport/
SetupIPMonitoringFromVPC_*"
            ],
            "Effect": "Allow"
        },
        {
            "Action": [
                "iam:DetachRolePolicy",
                "iam:AttachRolePolicy"
            ],
            "Resource": [
                "arn:aws:iam::aws:policy/service-role/AmazonSSMManagedInstanceCore"
            ],
            "Effect": "Allow"
        },
        {
            "Action": [
                "cloudwatch:DeleteDashboards"
            ],
            "Resource": [
                "*"
            ],
            "Effect": "Allow"
        },
        {
            "Action": [
```

```
                "ec2:AuthorizeSecurityGroupEgress",
                "ec2:CreateSecurityGroup",
                "ec2:DeleteSecurityGroup",
                "ec2:DescribeSubnets"
            ],
            "Resource": [
                "*"
            ],
            "Effect": "Allow"
        }
    ]
}
```

Étapes de document

1. aws:executeAwsApi : décrivez le sous-réseau fourni.
2. aws:branch : évaluez l'entrée.TargetIPs

   (IPv6) Si TargetIPs contient un IPv6 :

   aws:assertAwsResourceProperty : vérifiez que le sous-réseau fourni est associé à un poolIPv6
3. aws:executeAwsApi : obtenez la dernière AMI Amazon Linux 2 depuis Parameter Store.
4. aws:executeAwsApi : créez un groupe de sécurité pour le test dans le VPC du sous-réseau.

   (Nettoyage) Si la création du groupe de sécurité échoue :

   aws:executeAwsApi : supprimez le groupe de sécurité créé par l'automatisation, s'il existe.
5. aws:executeAwsApi : autorisez tout le trafic sortant dans le groupe de sécurité de test.

   (Nettoyage) Si la création de la règle de sortie du groupe de sécurité échoue :

   aws:executeAwsApi : supprimez le groupe de sécurité créé par l'automatisation, s'il existe.
6. aws:executeAwsApi – Créer un rôle IAM pour l'instance EC2 de test.

   (Nettoyage) Si la création du rôle échoue :
   a. aws:executeAwsApi : supprimez le rôle IAM créé par l'automatisation, s'il existe.
   b. aws:executeAwsApi : supprimez le groupe de sécurité créé par l'automatisation, s'il existe.
7. aws:executeAwsApi : attachez la stratégie AmazonSSMManagedInstanceCore gérée.

   (Nettoyage) Si l'attachement de la stratégie échoue :
   a. aws:executeAwsApi : détachez la stratégie AmazonSSMManagedInstanceCore gérée du rôle créé par l'automatisation, si elle est attachée.
   b. aws:executeAwsApi : supprimez le rôle IAM créé par l'automatisation.
   c. aws:executeAwsApi : supprimez le groupe de sécurité créé par l'automatisation, s'il existe.
8. aws:executeAwsApi : attachez une stratégie en ligne pour autoriser la définition des rétentions des groupes de journaux CloudWatch et la création d'un tableau de bord CloudWatch.

   (Nettoyage) Si l'attachement de la stratégie en ligne échoue :
   a. aws:executeAwsApi : supprimez la stratégie en ligne CloudWatch du rôle créé par l'automatisation, si elle a été créée.
   b. aws:executeAwsApi : détachez la stratégie AmazonSSMManagedInstanceCore gérée du rôle créé par l'automatisation.
   c. aws:executeAwsApi : supprimez le rôle IAM créé par l'automatisation.
   d. aws:executeAwsApi : supprimez le groupe de sécurité créé par l'automatisation, s'il existe.
9. aws:executeAwsApi : créez un profil d'instance IAM.

(Nettoyage) Si la création du profil d'instance échoue :

a. aws:executeAwsApi : supprimez le profil d'instance IAM créé par l'automatisation, s'il existe.
b. aws:executeAwsApi : supprimez la stratégie en ligne CloudWatch du rôle créé par l'automatisation.
c. aws:executeAwsApi : supprimez la stratégie gérée du rôle créé par l'automatisation.AmazonSSMManagedInstanceCore
d. aws:executeAwsApi : supprimez le rôle IAM créé par l'automatisation.
e. aws:executeAwsApi : supprimez le groupe de sécurité créé par l'automatisation, s'il existe.

10. aws:executeAwsApi : associez le profil d'instance IAM au rôle IAM.

(Nettoyage) Si l'association du profil d'instance et du rôle échoue :

a. aws:executeAwsApi : supprimez le profil d'instance IAM du rôle, s'il est associé.
b. aws:executeAwsApi : supprimez le profil d'instance IAM créé par l'automatisation.
c. aws:executeAwsApi : supprimez la stratégie en ligne CloudWatch du rôle créé par l'automatisation.
d. aws:executeAwsApi : détachez la stratégie AmazonSSMManagedInstanceCore gérée du rôle créé par l'automatisation.
e. aws:executeAwsApi : supprimez le rôle IAM créé par l'automatisation.
f. aws:executeAwsApi : supprimez le groupe de sécurité créé par l'automatisation, s'il existe.

11. aws:sleep : attendez que le profil d'instance devienne disponible.
12. aws:runInstances : créez l'instance de test dans le sous-réseau spécifié, en attachant le profil d'instance créé précédemment.

(Nettoyage) Si l'étape échoue :

a. aws:changeInstanceState : mettez fin à l'instance de test.
b. aws:executeAwsApi : supprimez le profil d'instance IAM du rôle.
c. aws:executeAwsApi : supprimez le profil d'instance IAM créé par l'automatisation.
d. aws:executeAwsApi : supprimez la stratégie en ligne CloudWatch du rôle créé par l'automatisation.
e. aws:executeAwsApi : détachez la stratégie AmazonSSMManagedInstanceCore gérée du rôle créé par l'automatisation.
f. aws:executeAwsApi : supprimez le rôle IAM créé par l'automatisation.
g. aws:executeAwsApi : supprimez le groupe de sécurité créé par l'automatisation, s'il existe.

13. aws:branch : évaluez l'entrée.TargetIPs

(IPv6) Si TargetIPs contient un IPv6 :

aws:executeAwsApi : affectez un IPv6 à l'instance de test.

14. aws:waitForAwsResourceProperty : attendez que l'instance de test devienne une instance gérée.

(Nettoyage) Si l'étape échoue :

a. aws:changeInstanceState : mettez fin à l'instance de test.
b. aws:executeAwsApi : supprimez le profil d'instance IAM du rôle.
c. aws:executeAwsApi : supprimez le profil d'instance IAM créé par l'automatisation.
d. aws:executeAwsApi : supprimez la stratégie en ligne CloudWatch du rôle créé par l'automatisation.
e. aws:executeAwsApi : détachez la stratégie AmazonSSMManagedInstanceCore gérée du rôle créé par l'automatisation.
f. aws:executeAwsApi : supprimez le rôle IAM créé par l'automatisation.
g. aws:executeAwsApi : supprimez le groupe de sécurité créé par l'automatisation, s'il existe.

15. aws:runCommand : installez les prérequis du test :

a. aws:changeInstanceState : mettez fin à l'instance de test.
b. aws:executeAwsApi : supprimez le profil d'instance IAM du rôle.
c. aws:executeAwsApi : supprimez le profil d'instance IAM créé par l'automatisation.
d. aws:executeAwsApi : supprimez la stratégie en ligne CloudWatch du rôle créé par l'automatisation.
e. aws:executeAwsApi : détachez la stratégie AmazonSSMManagedInstanceCore gérée du rôle créé par l'automatisation.
f. aws:executeAwsApi : supprimez le rôle IAM créé par l'automatisation.
g. aws:executeAwsApi : supprimez le groupe de sécurité créé par l'automatisation, s'il existe.

16. aws:runCommand : validez que les valeurs fournies IPs sont syntaxiquement correctes IPv4 et/ou IPv6 aux adresses :

(Nettoyage) Si l'étape échoue :

a. aws:changeInstanceState : mettez fin à l'instance de test.
b. aws:executeAwsApi : supprimez le profil d'instance IAM du rôle.
c. aws:executeAwsApi : supprimez le profil d'instance IAM créé par l'automatisation.
d. aws:executeAwsApi : supprimez la stratégie en ligne CloudWatch du rôle créé par l'automatisation.
e. aws:executeAwsApi : détachez la stratégie AmazonSSMManagedInstanceCore gérée du rôle créé par l'automatisation.
f. aws:executeAwsApi : supprimez le rôle IAM créé par l'automatisation.
g. aws:executeAwsApi : supprimez le groupe de sécurité créé par l'automatisation, s'il existe.

17. aws:runCommand : définissez le test MTR pour chacune des .IPs

(Nettoyage) Si l'étape échoue :

a. aws:changeInstanceState : mettez fin à l'instance de test.
b. aws:executeAwsApi : supprimez le profil d'instance IAM du rôle.
c. aws:executeAwsApi : supprimez le profil d'instance IAM créé par l'automatisation.
d. aws:executeAwsApi : supprimez la stratégie en ligne CloudWatch du rôle créé par l'automatisation.
e. aws:executeAwsApi : détachez la stratégie AmazonSSMManagedInstanceCore gérée du rôle créé par l'automatisation.
f. aws:executeAwsApi : supprimez le rôle IAM créé par l'automatisation.
g. aws:executeAwsApi : supprimez le groupe de sécurité créé par l'automatisation, s'il existe.

18. aws:runCommand : définissez le premier test ping pour chacun des IPs .

(Nettoyage) Si l'étape échoue :

a. aws:changeInstanceState : mettez fin à l'instance de test.
b. aws:executeAwsApi : supprimez le profil d'instance IAM du rôle.
c. aws:executeAwsApi : supprimez le profil d'instance IAM créé par l'automatisation.
d. aws:executeAwsApi : supprimez la stratégie en ligne CloudWatch du rôle créé par l'automatisation.
e. aws:executeAwsApi : détachez la stratégie AmazonSSMManagedInstanceCore gérée du rôle créé par l'automatisation.
f. aws:executeAwsApi : supprimez le rôle IAM créé par l'automatisation.
g. aws:executeAwsApi : supprimez le groupe de sécurité créé par l'automatisation, s'il existe.

19. aws:runCommand : définissez le deuxième test ping pour chacun des .IPs

(Nettoyage) Si l'étape échoue :

a. aws:changeInstanceState : mettez fin à l'instance de test.
b. aws:executeAwsApi : supprimez le profil d'instance IAM du rôle.
c. aws:executeAwsApi : supprimez le profil d'instance IAM créé par l'automatisation.
d. aws:executeAwsApi : supprimez la stratégie en ligne CloudWatch du rôle créé par l'automatisation.

e. aws:executeAwsApi : détachez la stratégie AmazonSSMManagedInstanceCore gérée du rôle créé par l'automatisation.
f. aws:executeAwsApi : supprimez le rôle IAM créé par l'automatisation.
g. aws:executeAwsApi : supprimez le groupe de sécurité créé par l'automatisation, s'il existe.

20. aws:runCommand : définissez le test tracepath pour chacun des .IPs

(Nettoyage) Si l'étape échoue :

a. aws:changeInstanceState : mettez fin à l'instance de test.
b. aws:executeAwsApi : supprimez le profil d'instance IAM du rôle.
c. aws:executeAwsApi : supprimez le profil d'instance IAM créé par l'automatisation.
d. aws:executeAwsApi : supprimez la stratégie en ligne CloudWatch du rôle créé par l'automatisation.
e. aws:executeAwsApi : détachez la stratégie AmazonSSMManagedInstanceCore gérée du rôle créé par l'automatisation.
f. aws:executeAwsApi : supprimez le rôle IAM créé par l'automatisation.
g. aws:executeAwsApi : supprimez le groupe de sécurité créé par l'automatisation, s'il existe.

21. aws:runCommand : définissez le test traceroute pour chacun des .IPs

(Nettoyage) Si l'étape échoue :

a. aws:changeInstanceState : mettez fin à l'instance de test.
b. aws:executeAwsApi : supprimez le profil d'instance IAM du rôle.
c. aws:executeAwsApi : supprimez le profil d'instance IAM créé par l'automatisation.
d. aws:executeAwsApi : supprimez la stratégie en ligne CloudWatch du rôle créé par l'automatisation.
e. aws:executeAwsApi : détachez la stratégie AmazonSSMManagedInstanceCore gérée du rôle créé par l'automatisation.
f. aws:executeAwsApi : supprimez le rôle IAM créé par l'automatisation.
g. aws:executeAwsApi : supprimez le groupe de sécurité créé par l'automatisation, s'il existe.

22. aws:runCommand : configurez les journaux CloudWatch.

(Nettoyage) Si l'étape échoue :

a. aws:changeInstanceState : mettez fin à l'instance de test.
b. aws:executeAwsApi : supprimez le profil d'instance IAM du rôle.
c. aws:executeAwsApi : supprimez le profil d'instance IAM créé par l'automatisation.
d. aws:executeAwsApi : supprimez la stratégie en ligne CloudWatch du rôle créé par l'automatisation.
e. aws:executeAwsApi : détachez la stratégie AmazonSSMManagedInstanceCore gérée du rôle créé par l'automatisation.
f. aws:executeAwsApi : supprimez le rôle IAM créé par l'automatisation.
g. aws:executeAwsApi : supprimez le groupe de sécurité créé par l'automatisation, s'il existe.

23. aws:runCommand : planifiez les tâches cron pour exécuter chaque test toutes les minutes.

(Nettoyage) Si l'étape échoue :

a. aws:changeInstanceState : mettez fin à l'instance de test.
b. aws:executeAwsApi : supprimez le profil d'instance IAM du rôle.
c. aws:executeAwsApi : supprimez le profil d'instance IAM créé par l'automatisation.
d. aws:executeAwsApi : supprimez la stratégie en ligne CloudWatch du rôle créé par l'automatisation.
e. aws:executeAwsApi : détachez la stratégie AmazonSSMManagedInstanceCore gérée du rôle créé par l'automatisation.
f. aws:executeAwsApi : supprimez le rôle IAM créé par l'automatisation.
g. aws:executeAwsApi : supprimez le groupe de sécurité créé par l'automatisation, s'il existe.

24. aws:sleep : attendez que les tests génèrent des données.

25. aws:runCommand : définissez les rétentions souhaitées du groupe de journaux CloudWatch.

    (Nettoyage) Si l'étape échoue :

    a. aws:changeInstanceState : mettez fin à l'instance de test.
    b. aws:executeAwsApi : supprimez le profil d'instance IAM du rôle.
    c. aws:executeAwsApi : supprimez le profil d'instance IAM créé par l'automatisation.
    d. aws:executeAwsApi : supprimez la stratégie en ligne CloudWatch du rôle créé par l'automatisation.
    e. aws:executeAwsApi : détachez la stratégie AmazonSSMManagedInstanceCore gérée du rôle créé par l'automatisation.
    f. aws:executeAwsApi : supprimez le rôle IAM créé par l'automatisation.
    g. aws:executeAwsApi : supprimez le groupe de sécurité créé par l'automatisation, s'il existe.
26. aws:runCommand : définissez les filtres de métriques du groupe de journaux CloudWatch.

    (Nettoyage) Si l'étape échoue :

    a. aws:changeInstanceState : mettez fin à l'instance de test.
    b. aws:executeAwsApi : supprimez le profil d'instance IAM du rôle.
    c. aws:executeAwsApi : supprimez le profil d'instance IAM créé par l'automatisation.
    d. aws:executeAwsApi : supprimez la stratégie en ligne CloudWatch du rôle créé par l'automatisation.
    e. aws:executeAwsApi : détachez la stratégie AmazonSSMManagedInstanceCore gérée du rôle créé par l'automatisation.
    f. aws:executeAwsApi : supprimez le rôle IAM créé par l'automatisation.
    g. aws:executeAwsApi : supprimez le groupe de sécurité créé par l'automatisation, s'il existe.
27. aws:runCommand : créez le tableau de bord CloudWatch.

    (Nettoyage) Si l'étape échoue :

    a. aws:executeAwsApi : supprimez le tableau de bord CloudWatch, s'il existe.
    b. aws:changeInstanceState : mettez fin à l'instance de test.
    c. aws:executeAwsApi : supprimez le profil d'instance IAM du rôle.
    d. aws:executeAwsApi : supprimez le profil d'instance IAM créé par l'automatisation.
    e. aws:executeAwsApi : supprimez la stratégie en ligne CloudWatch du rôle créé par l'automatisation.
    f. aws:executeAwsApi : détachez la stratégie AmazonSSMManagedInstanceCore gérée du rôle créé par l'automatisation.
    g. aws:executeAwsApi : supprimez le rôle IAM créé par l'automatisation.
    h. aws:executeAwsApi : supprimez le groupe de sécurité créé par l'automatisation, s'il existe.

Résultats

createCloudWatchDashboards.Output : URL du CloudWatch tableau de bord.

createManagedInstance.InstanceIds - ID de l'instance de test.

AWS-SetupManagedInstance

Description

Configurez une instance avec un rôle AWS Identity and Access Management (IAM) pour l'accès Systems Manager.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- InstanceId

  Type: String (chaîne)

  Description : (Obligatoire) ID de l'instance EC2 à configurer
- LambdaAssumeRole

  Type: String (chaîne)

  Description : (Facultatif) ARN du rôle qui autorise Lambda créé par Automation à effectuer les actions en votre nom. Si vous ne spécifiez pas cette valeur, un rôle transitoire est créé pour exécuter la fonction Lambda.
- RoleName

  Type: String (chaîne)

  Par défaut : SSMRoleForManagedInstance

  Description : (Facultatif) Nom du IAM rôle pour l'instance EC2. Si ce rôle n'existe pas, il est créé. Lorsque vous spécifiez cette valeur, vérifiez que le rôle contient la stratégie AmazonSSMManagedInstanceCore gérée.

AWS-SetupManagedRoleOnEC2Instance

Description

Configurez une instance avec le SSMRoleForManagedInstance rôle IAM géré pour lSystems Manager'accès.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- InstanceId

  Type: String (chaîne)

  Description : (Obligatoire) ID de l'instance EC2 à configurer
- LambdaAssumeRole

  Type: String (chaîne)

  Description : (Facultatif) ARN du rôle qui autorise Lambda créé par Automation à effectuer les actions en votre nom. Si vous ne spécifiez pas cette valeur, un rôle transitoire est créé pour exécuter la fonction Lambda.
- RoleName

  Type: String (chaîne)

  Par défaut : SSMRoleForManagedInstance

  Description : (Facultatif) Nom du rôle IAM pour l'instance EC2. Si ce rôle n'existe pas, il est créé. Lorsque vous spécifiez cette valeur, vérifiez que le rôle contient la stratégie AmazonSSMManagedInstanceCore gérée.

AWSSupport-ShareRDSSnapshot

Description

Le `AWSSupport-ShareRDSSnapshot` document fournit une solution automatisée pour la procédure décrite dans l'article du Centre de connaissances : Comment puis-je partager un instantané de base de données Amazon RDS chiffré avec un autre compte ? Si votre instantané Amazon Relational Database Service (Amazon RDS) a été chiffré à l'aide de la clé par défaut AWS Key Management Service (AWS KMS), vous ne pouvez pas partager l'instantané. Dans ce cas, vous devez copier l'instantané à l'aide d'une clé principale client (CMK), puis partager l'instantané avec le compte cible. Cette automatisation effectue ces étapes en utilisant la valeur que vous spécifiez dans le `SnapshotName` paramètre ou le dernier instantané trouvé pour l'instance ou le cluster Amazon RDS de base de données sélectionné.

> **Note**
>
> Si vous ne spécifiez pas de valeur pour le `KMSKey` paramètre, Automation crée une nouvelle AWS KMS clé CMK dans votre compte qui est utilisée pour chiffrer l'instantané.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Bases de données

Paramètres

- AccountIds

  Type: StringList

  Description : (Obligatoire) Liste séparée par des virgules du compte avec IDs lequel partager l'instantané.
- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- Database

  Type: String (chaîne)

  Description : (Obligatoire) Nom de l'instance ou du cluster Amazon RDS de base de données dont vous souhaitez partager l'instantané. Ce paramètre est facultatif si vous spécifiez une valeur pour le `SnapshotName` paramètre.
- KMSKey

  Type: String (chaîne)

  Description : (Facultatif) Amazon Resource Name (ARN) complet de la AWS KMS clé CMK utilisée pour chiffrer l'instantané.
- SnapshotName

  Type: String (chaîne)

  Description : (Facultatif) ID du cluster de bases de données ou de l'instantané d'instance que vous souhaitez utiliser.

Autorisations IAM nécessaires

`AutomationAssumeRole` nécessite les actions suivantes pour exécuter correctement le document Automation.

- `ssm:StartAutomationExecution`
- `rds:DescribeDBInstances`
- `rds:DescribeDBSnapshots`
- `rds:CopyDBSnapshot`
- `rds:ModifyDBSnapshotAttribute`

requiert les actions suivantes pour exécuter correctement le document Automation pour un cluster de bases de données.`AutomationAssumeRole`

- `ssm:StartAutomationExecution`
- `rds:DescribeDBClusters`
- `rds:DescribeDBClusterSnapshots`

- `rds:CopyDBClusterSnapshot`
- `rds:ModifyDBClusterSnapshotAttribute`

Le IAM rôle utilisé pour exécuter l'automatisation doit être ajouté en tant qu'utilisateur clé pour utiliser la AWS KMS clé CMK spécifiée dans le `ARNKmsKey` paramètre. Pour plus d'informations sur l'ajout d'utilisateurs de clé à une AWS KMS clé CMK, consultez Modification d'une stratégie de clé dans le AWS Key Management Service Developer Guide .

Le `AutomationAssumeRole` nécessite les actions supplémentaires suivantes pour exécuter correctement le document Automation si vous ne spécifiez pas de valeur pour le `KMSKey` paramètre.

- `kms:CreateKey`
- `kms:ScheduleKeyDeletion`

Étapes de document

1. aws:executeScript : vérifie si une valeur a été fournie pour le `KMSKey` paramètre et crée une AWS KMS clé CMK si aucune valeur n'est trouvée.
2. aws:branch : vérifie si une valeur a été fournie pour le `SnapshotName` paramètre et les branches en conséquence.
3. aws:executeAwsApi - Vérifie si l'instantané fourni provient d'une instance de base de données.
4. aws:executeScript - Formate le `SnapshotName` paramètre en remplaçant deux points par un trait d'union.
5. aws:executeAwsApi - Copie l'instantané à l'aide du `KMSKey` .
6. aws:waitForAwsResourceProperty : attend la fin de l'opération de copie de l'instantané.
7. aws:executeAwsApi - Partage le nouvel instantané avec l' `AccountIds` spécifié.
8. aws:executeAwsApi - Vérifie si l'instantané fourni provient d'un cluster de bases de données.
9. aws:executeScript - Formate le `SnapshotName` paramètre en remplaçant deux points par un trait d'union.
10. aws:executeAwsApi - Copie l'instantané à l'aide du `KMSKey` .
11. aws:waitForAwsResourceProperty : attend la fin de l'opération de copie de l'instantané.
12. aws:executeAwsApi - Partage le nouvel instantané avec l' `AccountIds` spécifié.
13. aws:executeAwsApi - Vérifie si la valeur fournie pour le `Database` paramètre est une instance de base de données.
14. aws:executeAwsApi - Vérifie si la valeur fournie pour le `Database` paramètre est un cluster de base de données.
15. aws:executeAwsApi - Récupère la liste des instantanés pour l' spécifié`Database`.
16. aws:executeScript : détermine le dernier instantané disponible à partir de la liste assemblée à l'étape précédente.
17. aws:executeAwsApi - Copie l'instantané d'instance de base de données à l'aide de l' spécifié`KMSKey`.
18. aws:waitForAwsResourceProperty : attend la fin de l'opération de copie de l'instantané.
19. aws:executeAwsApi - Partage le nouvel instantané avec l' `AccountIds` spécifié.
20. aws:executeAwsApi - Récupère la liste des instantanés pour l' spécifié`Database`.
21. aws:executeScript : détermine le dernier instantané disponible à partir de la liste assemblée à l'étape précédente.
22. aws:executeAwsApi - Copie l'instantané d'instance de base de données à l'aide de l' spécifié`KMSKey`.
23. aws:waitForAwsResourceProperty : attend la fin de l'opération de copie de l'instantané.
24. aws:executeAwsApi - Partage le nouvel instantané avec l' `AccountIds` spécifié.

25 aws:executeScript - Supprime la AWS KMS clé CMK créée par l'automatisation si vous n'avez pas spécifié de valeur pour le `KMSKey` paramètre et que l'automatisation échoue.

AWSEC2-SQLServerDBRestore

Description

Le document `AWSEC2-SQLServerDBRestore` restaure les sauvegardes de base de données Microsoft SQL Server stockées dans Amazon S3 pour SQL Server 2017 s'exécutant sur une instance Linux Amazon Elastic Compute Cloud (EC2). Vous pouvez fournir votre propre instance EC2 exécutant SQL Server 2017 Linux. Si une instance EC2 n'est pas fournie, le flux de travail Automation lance et configure une nouvelle instance EC2 Ubuntu 16.04 avec SQL Server 2017. Automation prend en charge la restauration des sauvegardes de journaux complètes, différentielles et transactionnelles. Cette instance d'Automation accepte plusieurs fichiers de sauvegarde de base de données et restaure automatiquement la dernière sauvegarde de chaque base de données valides dans les fichiers fournis.

Pour automatiser la sauvegarde et la restauration d'une base de données SQL Server sur site vers une instance EC2 exécutant SQL Server 2017 Linux, vous pouvez utiliser le AWS script signé par PowerShell `MigrateSQLServerToEC2Linux`.

> **Important**
>
> Ce flux de travail Automation réinitialise le mot de passe de l'utilisateur administrateur (SA) du serveur SQL Server à chaque exécution du flux de travail. Une fois le flux de travail Automation terminé, vous devez à nouveau définir votre propre mot de passe de l'utilisateur SA avant de vous connecter à l'instance SQL Server.

Exécution de cette automatisation (console)

Types de document

Automation

Propriétaire

Amazon

Plateformes

Linux

Conditions préalables requises

- Ce document Automation ne fonctionne qu'avec les instances EC2 pour Linux exécutant SQL Server.
- Ce flux de travail Automation doit être exécuté par un utilisateur avec, au minimum, les autorisations désignées dans la section Required IAM Permissions ci-dessous.
- Si vous fournissez votre propre instance EC2 :
  - Configurez l'instance EC2 avec un profil d'instance AWS Identity and Access Management (IAM) auquel est attachée la stratégie gérée `AmazonSSMManagedInstanceCore`. Pour plus d'informations, consultez la section Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34).
  - Vérifiez que l'Agent SSM est installé sur votre instance EC2. Pour plus d'informations, consultez la section Installation et configuration de l'Agent SSM sur les instances EC2 pour Linux (p. 75).
  - Vérifiez que l'instance EC2 dispose de suffisamment d'espace disque libre pour télécharger et restaurer les sauvegardes SQL Server.

Restrictions

Cette instance d'Automation ne prend pas en charge la restauration vers SQL Server s'exécutant sur des instances EC2 pour Windows Server. Cette instance d'Automation ne restaure les sauvegardes de base de données qui sont compatibles avec SQL Server Linux 2017. Pour de plus amples informations, veuillez consulter Fonctionnalités et éditions de SQL Server 2017 prises en charge sur Linux.

Parameters

- DatabaseNames

  Type: String (chaîne)

  Description : (Facultatif) Liste séparée par des virgules des noms des bases de données à restaurer.
- DataDirectorySize

  Type: String (chaîne)

  Description : (Facultatif) Taille de volume (Gio) souhaitée du répertoire de données SQL Server pour la nouvelle instance EC2.

  Valeur par défaut : 100
- KeyPair

  Type: String (chaîne)

  Description : (Facultatif) Paire de clés à utiliser lors de la création de la nouvelle instance EC2.
- IamInstanceProfileName

  Type: String (chaîne)

  Description : (Facultatif) Profil IAM d'instance à attacher à la nouvelle instance EC2. Le profil d'instance IAM doit être attaché à la stratégie gérée `AmazonSSMManagedInstanceCore`.
- InstanceId

  Type: String (chaîne)

  Description : (Facultatif) Instance exécutant SQL Server 2017 sous Linux. Si aucun nInstanceId'est fourni, l'automatisation lance une nouvelle instance EC2 à l'aide de l' InstanceType et SQLServerEdition des fournis.
- InstanceType

  Type: String (chaîne)

  Description : (Facultatif) Type d'instance de l'instance EC2 à lancer.
- IsS3PresignedUrl

  Type: String (chaîne)

  Description : (Facultatif) Si S3Input est une URL S3 pré-signée, indiquez `yes` .

  Valeur par défaut : non

  Valeurs valides : oui | non
- LogDirectorySize

  Type: String (chaîne)

  Description : (Facultatif) Taille de volume (Gio) souhaitée du répertoire des journaux SQL Server pour la nouvelle instance EC2.

Valeur par défaut : 100

- S3Input

  Type: String (chaîne)

  Description : (Obligatoire) Nom du compartiment S3, liste séparée par des virgules des clés d'objet S3 ou liste séparée par des virgules des S3 pré-signées URLs contenant les fichiers de sauvegarde SQL à restaurer.

- SQLServerEdition

  Type: String (chaîne)

  Description : (Facultatif) Édition de SQL Server 2017 à installer sur l'instance EC2 nouvellement créée.

  Valeurs valides : Standard | Enterprise | Web | Express

- SubnetId

  Type: String (chaîne)

  Description : (Facultatif) Sous-réseau dans lequel lancer la nouvelle instance EC2. Le sous-réseau doit disposer d'une connectivité sortante aux services AWS. Si aucune valeur n'SubnetIdest fournie pour , Automation utilise le sous-réseau par défaut.

- TempDbDirectorySize

  Type: String (chaîne)

  Description : (Facultatif) Taille de volume (Gio) souhaitée du TempDB répertoire SQL Server pour la nouvelle instance EC2.

  Valeur par défaut : 100

Autorisations IAM nécessaires

`AutomationAssumeRole` nécessite les actions suivantes pour exécuter correctement le document Automation.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeImages",
                "ec2:RunInstances",
                "ec2:CreateTags",
                "ec2:DescribeInstances",
                "ec2:DescribeInstanceStatus",
                "ec2:RebootInstances",
                "ssm:SendCommand",
                "ssm:GetAutomationExecution",
                "ssm:ListCommands",
                "ssm:StartAutomationExecution",
                "ssm:DescribeInstanceInformation",
                "ssm:ListCommandInvocations",
                "iam:PassRole"
            ],
            "Resource": "*"
        }
    ]
```

```
}
```

Étapes de document

Pour les nouvelles instances EC2 :

1. aws:executeAwsApi - Récupérez l'ID d'AMI pour SQL Server 2017 sur Ubuntu 16.04.
2. aws:runinstances - Lancez une nouvelle instance EC2 pour Linux.
3. aws:waitForAwsResourceProperty - Attendez que l'instance EC2 qui vient d'être créée soit prête.
4. aws:executeAwsApi - Redémarrez l'instance si l'instance n'est pas prête.
5. aws:assertAwsResourceProperty - Vérifiez que l'Agent SSM est installé.
6. aws:runCommand : Exécutez le script de restauration SQL Server dans PowerShell .

Pour les instances EC2 existantes :

1. aws:waitForAwsResourceProperty - Vérifiez que l'instance EC2 est prête.
2. aws:executeAwsApi - Redémarrez l'instance si l'instance n'est pas prête.
3. aws:assertAwsResourceProperty - Vérifiez que l'Agent SSM est installé.
4. aws:runCommand : Exécutez le script de restauration SQL Server dans PowerShell .

Résultats

getInstanceID d'instance

.SortierestoreToNewInstance

.SortierestoreToExistingInstance

AWS-StartEC2Instance

Description

Démarrez une ou plusieurs instances EC2.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.

- InstanceId

  Type: StringList

  Description : (Obligatoire) Instances EC2 à démarrer.

AWSSupport-StartEC2RescueWorkflow

Description

Le document d'Automation AWSSupport -StartEC2RescueWorkflow exécute le script encodé en base64 fourni (Bash ou Powershell) sur une instance d'assistant créée pour corriger votre instance. Le volume racine de votre instance est attaché et monté sur l'instance d'assistant, également connu sous le nom EC2Rescue d'instance. Si votre instance est basée sur Windows, fournissez un script Powershell. Dans le cas contraire, utilisez Bash. Le flux de travail définit des variables d'environnement que vous pouvez utiliser dans votre script. Les variables d'environnement contiennent des informations sur l'entrée que vous avez fournie, ainsi que des informations sur le volume racine hors connexion. Le volume hors connexion est déjà monté et prêt à être utilisé. Par exemple, vous pouvez enregistrer un fichier Desired State Configuration dans un volume racine Windows hors connexion ou exécuter une commande chroot pour un volume racine Linux hors connexion et effectuer une correction hors connexion.

Exécution de cette automatisation (console)

> **Important**
>
> Les instances Amazon EC2 créées à partir des AMI (Amazon Machine Images) Marketplace ne sont pas prises en charge par cette automatisation.

Informations supplémentaires

Pour coder un script en base64, vous pouvez utiliser Powershell ou Bash. PowerShell :

```
[System.Convert]::ToBase64String([System.Text.Encoding]::Unicode.GetBytes([System.IO.File]::ReadAllText
```

Bash :

```
base64 PATH_TO_FILE
```

Voici une liste des variables d'environnement que vous pouvez utiliser dans vos scripts hors connexion, en fonction du système d'exploitation cible

Windows:

Variable	Description	Exemple de valeur
$env:EC2RESCUE_ACCOUNT_ID	{{ global:ACCOUNT_ID }}	123456789012
$env:EC2RESCUE_DATE	{{ global:DATE }}	2018-09-07
$env:EC2RESCUE_DATE_TIME	{{ global:DATE_TIME }}	2018-09-07_18.09.59
$env:EC2RESCUE_EC2RW_DIR	EC2Rescuepour le chemin d'installation Windows	C:\Program Files\Amazon \EC2Rescue

Variable	Description	Exemple de valeur
$env:EC2RESCUE_EC2RW_DIR	EC2Rescuepour le chemin d'installation Windows	C:\Program Files\Amazon \EC2Rescue
$env:EC2RESCUE_EXECUTION_ID	{{ automation:EXECUTION_ID }}	7ef8008e-219b-4aca-8bb5-65e2e898e20b
$env:EC2RESCUE_OFFLINE_CURRENT_CONTROL_SET	Chemin de l'ensemble de contrôles Windows hors connexion	HKLM:\AWSTempSystem \ControlSet001
$env:EC2RESCUE_OFFLINE_DRIVE	Lecteur Windows hors connexion	D:\
$env:EC2RESCUE_OFFLINE_EBS_DEVICE	Périphérique EBS de volume racine hors connexion	xvdf
$env:EC2RESCUE_OFFLINE_KERNEL_VER	Version de noyau Windows hors connexion	6.1.7601.24214
$env:EC2RESCUE_OFFLINE_OS_ARCHITECTURE	Architecture Windows hors connexion	AMD64
$env:EC2RESCUE_OFFLINE_OS_CAPTION	Légende Windows hors connexion	Windows Server 2008 R2 Datacenter
$env:EC2RESCUE_OFFLINE_OS_TYPE	Type de système d'exploitation Windows hors connexion	de bases de données
$env:EC2RESCUE_OFFLINE_PROGRAM_FILES_DIR	Chemin du répertoire de fichiers de programme Windows hors connexion	D:\Program Files
$env:EC2RESCUE_OFFLINE_PROGRAM_FILES_X86_DIR	Chemin du répertoire de fichiers de programme x86 hors connexion	D:\Program Files (x86)
$env:EC2RESCUE_OFFLINE_REGISTRY_DIR	Chemin du répertoire de registre Windows hors connexion	D:\Windows\System32\config
$env:EC2RESCUE_OFFLINE_SYSTEM_ROOT	Chemin du répertoire racine du système Windows hors connexion	D:\Windows
$env:EC2RESCUE_REGION	{{ global:REGION }}	us-west-1
$env:EC2RESCUE_S3_BUCKET	{{ S3BucketName }}	mybucket
$env:EC2RESCUE_S3_PREFIX	{{ S3Prefix }}	myprefix/
$env:EC2RESCUE_SOURCE_INSTANCE	{{ InstanceId }}	i-abcdefgh123456789
$script:EC2RESCUE_OFFLINE_WINDOWS_INSTALL	Windows Installation Windows hors connexion	Objet Powershell client

Linux :

Variable	Description	Exemple de valeur
EC2RESCUE_ACCOUNT_ID	{{ global:ACCOUNT_ID }}	123456789012

Variable	Description	Exemple de valeur
EC2RESCUE_DATE	{{ global:DATE }}	2018-09-07
EC2RESCUE_DATE_TIME	{{ global:DATE_TIME }}	2018-09-07_18.09.59
EC2RESCUE_EC2RL_DIR	EC2Rescuepour le chemin d'installation Linux	/usr/local/ec2rl-1.1.3
EC2RESCUE_EXECUTION_ID	{{ automation:EXECUTION_ID }}	7ef8008e-219b-4aca-8bb5-65e2e898e20b
EC2RESCUE_OFFLINE_DEVICE	Nom du périphérique hors connexion	/dev/xvdf1
EC2RESCUE_OFFLINE_EBS_DEVICE	Périphérique EBS de volume racine hors connexion	/dev/sdf
EC2RESCUE_OFFLINE_SYSTEM_ROOT	Nom de montage du volume racine hors connexion	/mnt/mount
EC2RESCUE_PYTHON	Version de Python	python2.7
EC2RESCUE_REGION	{{ global:REGION }}	us-west-1
EC2RESCUE_S3_BUCKET	{{ S3BucketName }}	mybucket
EC2RESCUE_S3_PREFIX	{{ S3Prefix }}	myprefix/
EC2RESCUE_SOURCE_INSTANCE	{{ InstanceId }}	i-abcdefgh123456789

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AMIPrefix

  Type: String (chaîne)

  Par défaut : AWSSupport-EC2Rescue

  Description : (Facultatif) Un préfixe pour le nom de l'AMI de sauvegarde.
- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- CreatePostEC2RescueBackup

  Type: String (chaîne)

Valeurs valides : Vrai | Faux

Par défaut : Faux

Description : (Facultatif) Définissez ce paramètre sur True pour créer une AMI d' InstanceId après l'exécution du script, avant de la démarrer. L'AMI sera conservée une fois l'automatisation terminée. Vous devez sécuriser l'accès à l'AMI ou le supprimer.

- CreatePreEC2RescueBackup

  Type: String (chaîne)

  Valeurs valides : Vrai | Faux

  Par défaut : Faux

  Description : (Facultatif) Définissez ce paramètre sur True pour créer une AMI d' InstanceId avant d'exécuter le script. L'AMI sera conservée une fois l'automatisation terminée. Vous devez sécuriser l'accès à l'AMI ou le supprimer.
- EC2RescueInstanceType

  Type: String (chaîne)

  Valeurs valides : t2.small | t2.medium | t2.large

  Par défaut : t2.small

  Description : (Facultatif) Type d'instance EC2 de l'EC2Rescueinstance.
- InstanceId

  Type: String (chaîne)

  Description : (Obligatoire) ID de votre instance EC2. IMPORTANT :AWS Systems Manager Automation arrête cette instance. Les données stockées sur les volumes de stockage d'instance seront perdues. L'adresse IP publique change si vous n'utilisez pas une adresse IP Elastic.
- OfflineScript

  Type: String (chaîne)

  Description : (Obligatoire) Script codé en Base64 à exécuter sur l'instance d'assistant. Utilisez Bash si votre instance source est Linux, et PowerShell si elle est Windows.
- S3BucketName

  Type: String (chaîne)

  Description : (Facultatif) Nom du compartiment S3 de votre compte dans lequel vous souhaitez charger les journaux de dépannage. Assurez-vous que la stratégie de compartiment n'accorde pas des autorisations en lecture/écriture superflues pour les tiers qui n'ont pas besoin d'accéder aux journaux collectés.
- S3Prefix

  Type: String (chaîne)

  Par défaut : AWSSupport-EC2Rescue

  Description : (Facultatif) Préfixe pour les journaux S3.
- SubnetId

  Type: String (chaîne)

Par défaut : SelectedInstanceSubnet

Description : (Facultatif) ID de sous-réseau de l'EC2Rescueinstance. Par défaut, le même sous-réseau dans lequel l'instance réside est utilisé. IMPORTANT : Si vous fournissez un sous-réseau personnalisé, il doit être dans la même zone de disponibilité qu' et doit autoriser l'accès aux points de terminaison SSM.InstanceId

- UniqueId

  Type: String (chaîne)

  Par défaut : {{ automation:EXECUTION_ID }}

  Description : (Facultatif) Identifiant unique du flux de travail.

Autorisations IAM nécessaires

`AutomationAssumeRole` nécessite les actions suivantes pour exécuter correctement le document Automation.

Il est recommandé que la stratégie gérée AmazonSSMAutomationRole IAM soit attachée à l'utilisateur qui exécute l'automatisation. En plus de cette stratégie, l'utilisateur doit avoir :

```
{
                    "Version": "2012-10-17",
                    "Statement": [
                       {
                          "Action": [
                             "lambda:InvokeFunction",
                             "lambda:DeleteFunction",
                             "lambda:GetFunction"
                          ],
                          "Resource": "arn:aws:lambda:*:An-AWS-Account-
ID:function:AWSSupport-EC2Rescue-*",
                          "Effect": "Allow"
                       },
                       {
                          "Action": [
                             "s3:GetObject",
                             "s3:GetObjectVersion"
                          ],
                          "Resource": [
                             "arn:aws:s3:::awssupport-ssm.*/*.template",
                             "arn:aws:s3:::awssupport-ssm.*/*.zip"
                          ],
                          "Effect": "Allow"
                       },
                       {
                          "Action": [
                             "iam:CreateRole",
                             "iam:CreateInstanceProfile",
                             "iam:GetRole",
                             "iam:GetInstanceProfile",
                             "iam:PutRolePolicy",
                             "iam:DetachRolePolicy",
                             "iam:AttachRolePolicy",
                             "iam:PassRole",
                             "iam:AddRoleToInstanceProfile",
                             "iam:RemoveRoleFromInstanceProfile",
                             "iam:DeleteRole",
                             "iam:DeleteRolePolicy",
                             "iam:DeleteInstanceProfile"
```

```
                            ],
                            "Resource": [
                               "arn:aws:iam::An-AWS-Account-ID:role/AWSSupport-EC2Rescue-*",
                               "arn:aws:iam::An-AWS-Account-ID:instance-profile/AWSSupport-
EC2Rescue-*"
                            ],
                            "Effect": "Allow"
                         },
                         {
                            "Action": [
                               "lambda:CreateFunction",
                               "ec2:CreateVpc",
                               "ec2:ModifyVpcAttribute",
                               "ec2:DeleteVpc",
                               "ec2:CreateInternetGateway",
                               "ec2:AttachInternetGateway",
                               "ec2:DetachInternetGateway",
                               "ec2:DeleteInternetGateway",
                               "ec2:CreateSubnet",
                               "ec2:DeleteSubnet",
                               "ec2:CreateRoute",
                               "ec2:DeleteRoute",
                               "ec2:CreateRouteTable",
                               "ec2:AssociateRouteTable",
                               "ec2:DisassociateRouteTable",
                               "ec2:DeleteRouteTable",
                               "ec2:CreateVpcEndpoint",
                               "ec2:DeleteVpcEndpoints",
                               "ec2:ModifyVpcEndpoint",
                               "ec2:Describe*"
                            ],
                            "Resource": "*",
                            "Effect": "Allow"
                         }
                      ]
                   }
```

Étapes de document

1. aws:executeAwsApi : décrivez l'instance fournie.
2. aws:executeAwsApi : décrivez le volume racine de l'instance fournie.
3. aws:assertAwsResourceProperty : vérifiez que le type de périphérique du volume racine est EBS.
4. aws:assertAwsResourceProperty : vérifiez que le volume racine n'est pas chiffré.
5. aws:assertAwsResourceProperty : vérifiez l'ID de sous-réseau fourni.
   a. (Utiliser le sous-réseau de l'instance actuelle) - Si *SubnetId = *, exécutez aws:createStack pour déployer la SelectedInstanceSubnet EC2Rescue pileCloudFormation
   b. (Créer un nouveau VPC) - Si *SubnetId = *, exécutez aws:createStack pour déployer la CreateNewVPC EC2Rescue pileCloudFormation
   c. (Utiliser un sous-réseau personnalisé) - Dans tous les autres cas :

      aws:assertAwsResourceProperty : vérifiez que le sous-réseau fourni est dans la même zone de disponibilité que l'instance fournie.

      aws:createStack : déployez la EC2Rescue CloudFormation pile.
6. aws:invokeLambdaFunction : effectuez la validation d'entrée supplémentaire.
7. aws:executeAwsApi : mettez à jour la EC2Rescue CloudFormation pile pour créer l'instance EC2Rescue d'assistant.
8. aws:waitForAwsResourceProperty : attendez la fin de la mise à jour de la EC2Rescue CloudFormation pile.

9. aws:executeAwsApi : décrivez la sortie de la EC2Rescue CloudFormation pile pour obtenir l'ID de l'instance d'EC2Rescueassistant.
10. aws:waitForAwsResourceProperty : attendez que l'instance d'EC2Rescueassistant devienne une instance gérée.
11. aws:changeInstanceState : arrêtez l'instance fournie.
12. aws:changeInstanceState : arrêtez l'instance fournie.
13. aws:changeInstanceState : forcez l'arrêt de l'instance fournie.
14. aws:assertAwsResourceProperty : vérifiez la valeur CreatePreEC2RescueBackup d'entrée.
    a. (Créer une sauvegarde avant EC2Rescue) - Si *CreatePreEC2RescueBackup = True*
    b. aws:executeAwsApi : créez une sauvegarde d'AMI de l'instance fournie.
    c. aws:createTags : balisez la sauvegarde de l'AMI.
15. aws:runCommand : installez EC2Rescue sur l'instance EC2Rescue d'assistant
16. aws:executeAwsApi : détachez le volume racine de l'instance fournie.
17. aws:assertAwsResourceProperty : vérifiez la plateforme de l'instance fournie.
    a. (Instance Windows) :

       aws:executeAwsApi : attachez le volume racine à l'instance d'EC2Rescueassistant en tant que *xvdf*.

       aws:sleep : mettez en veille 10 secondes.

       aws:runCommand : exécutez le script hors connexion fourni dans Powershell.
    b. (Instance Linux) :

       aws:executeAwsApi : attachez le volume racine à l'instance d'EC2Rescueassistant en tant que */dev/sdf*.

       aws:sleep : mettez en veille 10 secondes.

       aws:runCommand : exécutez le script hors connexion fourni dans Bash.
18. aws:changeInstanceState : arrêtez l'instance EC2Rescue d'assistant.
19. aws:changeInstanceState : forcez l'arrêt de l'instance EC2Rescue d'assistant.
20. aws:executeAwsApi : détachez le volume racine de l'instance d'EC2Rescueassistant.
21. aws:executeAwsApi : rattachez le volume racine à l'instance fournie.
22. aws:assertAwsResourceProperty : vérifiez la valeur CreatePostEC2RescueBackup d'entrée.
    a. (Créer une sauvegarde après EC2Rescue) - Si *CreatePostEC2RescueBackup = True*
    b. aws:executeAwsApi : créez une sauvegarde d'AMI de l'instance fournie.
    c. aws:createTags : balisez la sauvegarde de l'AMI.
23. aws:executeAwsApi : restaurez l'état initial de suppression du volume racine de l'instance fournie
24. aws:changeInstanceState : restaurez l'état initial de l'instance fournie (en cours d'exécution ou arrêtée).
25. aws:deleteStack : supprimez la EC2Rescue CloudFormation pile.

Résultats

.SortierunScriptForLinux

.SortierunScriptForWindows

preScriptBackupID d'image

postScriptBackupID d'image

AWS-StartRdsInstance

Description

Démarrez une instance Amazon Relational Database Service (Amazon RDS).

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Bases de données

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- InstanceId

  Type: String (chaîne)

  Description : (Obligatoire) ID de l'Amazon RDSinstance à démarrer.

AWS-StopEC2Instance

Description

Arrêtez une ou plusieurs instances EC2.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.

- InstanceId

  Type: StringList

  Description : (Obligatoire) IDs d'une ou de plusieurs instances EC2 à arrêter

AWS-StopRdsInstance

Description

Arrêtez une instance Amazon Relational Database Service (Amazon RDS). Ce document appelle l'action StopDBInstance d'API . LStopDBInstance'API ne s'applique pas à Aurora MySQL et AuroraPostgreSQL. Pour les clusters Aurora, vous devez utiliser l'action StopDBCluster d'API à la place.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Bases de données

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- InstanceId

  Type: String (chaîne)

  Description : (Obligatoire) ID de l'Amazon RDSinstance à arrêter.

AWS-TerminateEC2Instance

Description

Supprimez une ou plusieurs instances EC2.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- InstanceId

  Type: StringList

  Description : (Obligatoire) IDs d'une ou de plusieurs instances EC2 à résilier.

AWSSupport-TerminateIPMonitoringDeVPC

Description

AWSSupport-TerminateIPMonitoringFromVPC met fin à un test de surveillance IP précédemment démarré par AWSSupport -SetupIPMonitoringFromVPC. Les données relatives à l'ID de test spécifié sont supprimées.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- InstanceId

  Type: String (chaîne)

  Description : (Obligatoire) ID de l'instance de surveillance.
- SubnetId

  Type: String (chaîne)

Description : (Obligatoire) ID de sous-réseau pour l'instance de surveillance.

Autorisations IAM nécessaires

`AutomationAssumeRole` nécessite les actions suivantes pour exécuter correctement le document Automation.

Il est recommandé que la stratégie AmazonSSMAutomationRole IAM gérée soit attachée à l'utilisateur qui exécute l'automatisation. En outre, l'utilisateur doit disposer de la stratégie suivante associée à son compte utilisateur, groupe ou rôle :

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Action": [
            "iam:DetachRolePolicy",
            "iam:RemoveRoleFromInstanceProfile",
            "iam:DeleteRole",
            "iam:DeleteInstanceProfile",
            "iam:DeleteRolePolicy"
         ],
         "Resource": [
            "arn:aws:iam::An-AWS-Account-ID:role/AWSSupport/SetupIPMonitoringFromVPC_*",
            "arn:aws:iam::An-AWS-Account-ID:instance-profile/AWSSupport/
SetupIPMonitoringFromVPC_*"
         ],
         "Effect": "Allow"
      },
      {
         "Action": [
            "iam:DetachRolePolicy"
         ],
         "Resource": [
            "arn:aws:iam::aws:policy/service-role/AmazonSSMManagedInstanceCore"
         ],
         "Effect": "Allow"
      },
      {
         "Action": [
            "cloudwatch:DeleteDashboards"
         ],
         "Resource": [
            "*"
         ],
         "Effect": "Allow"
      }
   ]
}
```

Étapes de document

1. aws:assertAwsResourceProperty : vérifiez AutomationExecutionId et InstanceId êtes lié au même test.
2. aws:assertAwsResourceProperty : vérifiez SubnetId et InstanceId êtes lié au même test.
3. aws:executeAwsApi : récupérez le groupe de sécurité du test.
4. aws:executeAwsApi : supprimez le tableau de bord CloudWatch.
5. aws:changeInstanceState : mettez fin à l'instance de test.
6. aws:executeAwsApi : supprimez le profil d'instance IAM du rôle.
7. aws:executeAwsApi : supprimez le profil d'instance IAM créé par l'automatisation.

8. aws:executeAwsApi : supprimez la stratégie en ligne CloudWatch du rôle créé par l'automatisation.
9. aws:executeAwsApi : détachez la stratégie AmazonSSMManagedInstanceCore gérée du rôle créé par l'automatisation.
10. aws:executeAwsApi : supprimez le rôle IAM créé par l'automatisation.
11. aws:executeAwsApi : supprimez le groupe de sécurité créé par l'automatisation, s'il existe.

Résultats

None (Aucun)

AWSSupport-TroubleshootConnectivityToRDS

Description

Le document AWSSupport -TroubleshootConnectivityToRDS Automation diagnostique les problèmes de connectivité entre une instance EC2 et une Amazon Relational Database Service instance. L'automatisation garantit que l'instance de base de données est disponible, puis vérifie les règles de groupe de sécurité associées, les listes de contrôle d'accès réseau (réseau) et les tables de routage pour détecter les problèmes de connectivité potentiels.ACLs

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- DBInstanceIdentifier

  Type: String (chaîne)

  Description : (Obligatoire) ID de l'instance de base de données à laquelle tester la connectivité.
- SourceInstance

  Type: String (chaîne)

  Modèle autorisé : ^i-[a-z0-9]{8,17}$

  Description : (Obligatoire) ID de l'instance EC2 à partir de laquelle tester la connectivité.

Autorisations IAM nécessaires

`AutomationAssumeRole` nécessite les actions suivantes pour exécuter correctement le document Automation.

- `ec2:DescribeInstances`
- `ec2:DescribeNetworkAcls`
- `ec2:DescribeRouteTables`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeSubnets`
- `rds:DescribeDBInstances`

Étapes de document

- aws:assertAwsResourceProperty - Confirme que l'état de l'instance DB est `available`.
- aws:executeAwsApi - Récupère des informations sur l'instance DB.
- aws:executeAwsApi - Récupère des informations sur le réseau d'instances de base de donnéesACLs.
- aws:executeAwsApi - Récupère le CIDR du sous-réseau de l'instance DB.
- aws:executeAwsApi - Récupère des informations sur l'instance EC2.
- aws:executeAwsApi - Récupère des informations sur le réseau d'instances EC2.ACLs
- aws:executeAwsApi - Récupère des informations sur les groupes de sécurité associés à l'instance EC2.
- aws:executeAwsApi - Récupère des informations sur les groupes de sécurité associés à l'instance DB.
- aws:executeAwsApi - Récupère des informations sur les tables de routage associées à l'instance EC2.
- aws:executeAwsApi - Récupère des informations sur la table de routage principale associée au Amazon VPC pour l'instance EC2.
- aws:executeAwsApi - Récupère des informations sur les tables de routage associées à l'instance DB.
- aws:executeAwsApi - Récupère des informations sur la table de routage principale associée au Amazon VPC pour l'instance DB.
- aws:executeScript - Évalue les règles du groupe de sécurité.
- aws:executeScript - Évalue le réseauACLs.
- aws:executeScript - Évalue les tables de routage.
- aws:sleep - Termine l'exécution de l'automatisation.

Résultats

getRDSInstanceProperties.DBInstanceIdentifier - Instance de base de données utilisée dans l'exécution de l'automatisation.

getRDSInstanceProperties.DBInstanceStatus - Statut actuel du DBInstance .

evalSecurityGroupRules.SecurityGroupEvaluation - Résultats de la comparaison des règles de groupe `SourceInstance` de sécurité aux règles de groupe de sécurité d'instance de base de données.

evalNetworkAclRules.NetworkAclEvaluation - Résultats de la comparaison du `SourceInstance` réseau ACLs avec le réseau de l'instance de base de donnéesACLs.

evalRouteTableEntries.RouteTableEvaluation - Résultats de la comparaison de la table `SourceInstance` de routage aux routes de l'instance de base de données.

AWSSupport-TroubleshootDirectoryTrust

Description

Le document AWSSupport Automation -TroubleshootDirectoryTrust diagnostique les problèmes de création d'approbation entre un AWS Managed Microsoft AD et un Microsoft Active Directory. L'automatisation

garantit que le type d'annuaire prend en charge les approbations, puis vérifie les règles de groupe de sécurité associées, les listes de contrôle d'accès réseau (réseau) et les tables de routage pour détecter les problèmes de connectivité potentiels.ACLs

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- DirectoryId

  Type: String (chaîne)

  Modèle autorisé : ^d-[a-z0-9]{10}$

  Description : (Obligatoire) ID de l' AWS Managed Microsoft AD à dépanner.
- RemoteDomainCidrs

  Type: StringList

  Modèle autorisé : ^(([0-9]|[1-9][0-9]|1[0-9]9]|2[0-4][0-9]|25[0-5])\\.{3}([[0-9]|[1-9][0-9]|1[0-9]9]|2[0-4][0-9]|25[0-5])(/(3[0-9]|[[[0-9]|9]]]{{{{{{]}}}}}{{{{{0-9]]{{{{{{{9]}}}}}{{{{{9]}}}}{{{{{{{9]}}}}}{{{{{{{]}}}}}{{{]{{{{]}}}}{{{]}{]}{]}}}{]}} à]]{]}{]{{ }} à]{}{}{}

- RequiredTrafficSG

  Type: String (chaîne)

  Description : (Obligatoire) Les exigences de port par défaut pour AWS Managed Microsoft AD . Dans la plupart des cas, vous ne devez pas modifier la valeur par défaut.

  Par défaut : {"inbound":{"tcp":[[53,53],[88,88],[135,135],[389,389],[445,445],[464,464],[636,636],[1024,65535]],"udp":[[53,53],[88,88],[123.123],[138,138],[389,389],[445,445],[464,464]],"icmp":[[-1,-1]]},"outbound":{"-1":[[0,65535]]}}
- TrustId

  Type: String (chaîne)

  Description : (Facultatif) ID de la relation d'approbation à dépanner.

Autorisations IAM nécessaires

`AutomationAssumeRole` nécessite les actions suivantes pour exécuter correctement le document Automation.

- `ds:DescribeConditionalForwarders`
- `ds:DescribeDirectories`
- `ds:DescribeTrusts`
- `ds:ListIpRoutes`
- `ec2:DescribeNetworkAcls`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeSubnets`

Étapes de document

- aws:assertAwsResourceProperty - Confirme que le type de répertoire est AWS Managed Microsoft AD.
- aws:executeAwsApi - Récupère des informations sur le AWS Managed Microsoft AD.
- aws:branch - Flux de travail des branches si une valeur est fournie pour le paramètre d'entrée `TrustId`.
- aws:executeAwsApi - Récupère des informations sur la relation d'approbation.
- aws:executeAwsApi - Récupère les adresses IP DNS du redirecteur conditionnel pour le `RemoteDomainName`.
- aws:executeAwsApi - Récupère des informations sur les routes IP qui ont été ajoutées à AWS Managed Microsoft AD.
- aws:executeAwsApi - Récupère ICIDRs' des AWS Managed Microsoft AD sous-réseaux.
- aws:executeAwsApi - Récupère des informations sur les groupes de sécurité associés à AWS Managed Microsoft AD.
- aws:executeAwsApi - Récupère des informations sur le réseau associé ACLs à .AWS Managed Microsoft AD
- aws:executeAwsApi - Confirme que les valeurs `RemoteDomainCidrs` sont valables. Confirme que le AWS Managed Microsoft AD a des redirecteurs conditionnels pour le `RemoteDomainCidrs`, et que les routes IP requises ont été ajoutées à AWS Managed Microsoft AD si les `RemoteDomainCidrs` sont des adresses IP non RFC 1918.
- aws:executeScript - Évalue les règles du groupe de sécurité.
- aws:executeScript - Évalue le réseauACLs.

Résultats

evalDirectorySecurityGroup.output - Résultats du fait d'évaluer si les règles de groupe de sécurité associées au AWS Managed Microsoft AD autorisent le trafic requis pour la création d'une approbation.

evalAclEntries.output - Résultats du fait d'évaluer si le réseau associé ACLs au AWS Managed Microsoft AD autorise le trafic requis pour la création d'une approbation.

evaluateRemoteDomainCidr.output - Résultats du fait d'évaluer si les `RemoteDomainCidrs` sont des valeurs valides. Confirme que le AWS Managed Microsoft AD a des redirecteurs conditionnels pour le `RemoteDomainCidrs`, et que les routes IP requises ont été ajoutées à AWS Managed Microsoft AD si les `RemoteDomainCidrs` sont des adresses IP non RFC 1918.

AWSSupport-TroubleshootRDP

Description

Le document d'Automation AWSSupport -TroubleshootRDP permet à l'utilisateur de vérifier ou de modifier les paramètres courants sur l'instance cible qui peuvent avoir un impact sur les connexions RDP (Remote Desktop Protocol), tels que le port RDP, l'authentification de couche réseau et les profils de pare-feu Windows. Le cas échéant, les modifications peuvent être appliquées hors connexion par l'arrêt et le redémarrage de l'instance, si l'utilisateur autorise explicitement la correction hors connexion. Par défaut, le document lit et produit les valeurs de ces paramètres.

> **Important**
>
> Les modifications apportées aux paramètres RDP, au service RDP et aux profils de pare-feu Windows doivent être vérifiées minutieusement avant d'exécuter ce document.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows

Paramètres

- Action

  Type: String (chaîne)

  Valeurs valides : CheckAll| FixAll | Personnalisé

  Par défaut : Personnalisée

  Description : [Facultatif] [Personnalisé] Utilisez les valeurs de Pare-feu, RDPServiceStartupType , , RDPServiceAction et RDPPortAction pour gérer les paramètres.NLASettingActionRemoteConnections [CheckAll] Lire les valeurs des paramètres sans les modifier. [FixAll] Restaurez les paramètres RDP par défaut et désactivez le pare-feu Windows.
- AllowOffline

  Type: String (chaîne)

  Valeurs valides : Vrai | Faux

  Par défaut : Faux

Description : (Facultatif) Correctif uniquement - Définissez ce paramètre sur true si vous autorisez la correction RDP hors connexion dans le cas où le dépannage en ligne échouerait ou dans le cas où l'instance fournie ne serait pas une instance gérée. Remarque: Pour la correction hors connexion, SSM Automation arrête l'instance et crée une AMI avant d'essayer toute autre opération.

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- Pare-feu

  Type: String (chaîne)

  Valeurs valides : Check | Disable (Vérifier | Désactiver)

  Par défaut : Check

  Description : (Facultatif) Vérifiez ou désactivez le pare-feu Windows (tous les profils).
- InstanceId

  Type: String (chaîne)

  Description : (Obligatoire) ID de l'instance dont les paramètres RDP doivent être résolus.
- NLASettingAction

  Type: String (chaîne)

  Valeurs valides : Check | Disable (Vérifier | Désactiver)

  Par défaut : Check

  Description : (Facultatif) Vérifiez ou désactivez l'authentification NLA (Network Layer Authentication).
- RDPPortAction

  Type: String (chaîne)

  Valeurs valides : Vérifier | Modifier

  Par défaut : Check

  Description : (Facultatif) Vérifiez le port utilisé pour les connexions RDP ou modifiez le port RDP sur 3389 et redémarrez le service.
- RDPServiceAction

  Type: String (chaîne)

  Valeurs valides : Check | Start | Restart | Force-Restart (Vérifier | Démarrer | Redémarrer | Forcer le redémarrage)

  Par défaut : Check

  Description : (Facultatif) Vérifiez, démarrez, redémarrez ou forcez le redémarrage du service RDP (TermService).
- RDPServiceStartupType

  Type: String (chaîne)

Valeurs valides : Check | Auto (Vérifier | Automatique)

Par défaut : Check

Description : (Facultatif) Vérifiez ou définissez le service RDP pour qu'il démarre automatiquement au démarrage de Windows.

- RemoteConnections

  Type: String (chaîne)

  Valeurs valides : Check | Enable (Vérifier | Activer)

  Par défaut : Check

  Description : (Facultatif) Une action à effectuer sur le fDenyTSConnections paramètre : Cochez, Activez.

- S3BucketName

  Type: String (chaîne)

  Description : (Facultatif) Offline only - Nom du compartiment S3 de votre compte dans lequel vous souhaitez charger les journaux de dépannage. Assurez-vous que la stratégie de compartiment n'accorde pas des autorisations en lecture/écriture superflues pour les tiers qui n'ont pas besoin d'accéder aux journaux collectés.

- SubnetId

  Type: String (chaîne)

  Par défaut : SelectedInstanceSubnet

  Description : (Facultatif) Offline only - ID de sous-réseau de l'EC2Rescueinstance utilisée pour effectuer le dépannage hors connexion. Si aucun ID de sous-réseau n'est spécifié, AWS Systems Manager Automation crée un VPC. IMPORTANT : Le sous-réseau doit être dans la même zone de disponibilité qu' InstanceId et doit autoriser l'accès aux points de terminaison SSM.

Autorisations IAM nécessaires

`AutomationAssumeRole` nécessite les actions suivantes pour exécuter correctement le document Automation.

Il est recommandé que l'instance EC2 recevant la commande dispose d'un rôle IAM avec la stratégie gérée AmazonSSMManagedInstanceCore Amazon attachée. Pour la correction en ligne, l'utilisateur doit avoir au moins ssm:DescribeInstanceInformation, ssm:ExecuteAutomation et ssm:SendCommand pour exécuter l'instance d'Automation et envoyer la commande à l'instance, plus ssm:GetAutomationExecution pour pouvoir lire la sortie d'Automation. Pour la correction hors connexion, l'utilisateur doit avoir au moins ssm:DescribeInstanceInformation, ssm:ExecuteAutomation, ec2:DescribeInstances, plus ssm:GetAutomationExecution pour pouvoir lire la sortie de l'automatisation. AWSSupport-TroubleshootRDP calls AWSSupport -ExecuteEC2Rescue to perform the offline remediation - please review the permissions for AWSSupport -ExecuteEC2Rescue to sure you can run the automation.

Étapes de document

1. aws:assertAwsResourceProperty : vérifiez s'il s'agit d'une instance Windows Server.
2. aws:assertAwsResourceProperty : vérifiez s'il s'agit d'une instance gérée.
3. (Dépannage en ligne) S'il s'agit d'une instance gérée :
   a. aws:assertAwsResourceProperty : vérifiez la valeur d'action fournie.
   b. (Vérification en ligne) Si Action = CheckAll , alors :

aws:runPowerShellScript : Exécute le PowerShell script pour obtenir le statut des profils de pare-feu Windows.

aws:executeAutomation : appelle AWSSupport -ManageWindowsService pour obtenir le statut du service RDP.

aws:executeAutomation : appelle AWSSupport -ManageRDPSettings pour obtenir les paramètres RDP.

c. (Correction en ligne) Si Action = FixAll , alors :

aws:runPowerShellScript : Exécute le PowerShell script pour désactiver tous les profils de pare-feu Windows.

aws:executeAutomation : appelle AWSSupport -ManageWindowsService pour démarrer le service RDP.

aws:executeAutomation - Appels -ManageRDPSettings pour activer les connexions à distance et désactiver NLA.AWSSupport

d. (Gestion en ligne) Si Action = Custom :

aws:runPowerShellScript : Exécute le PowerShell script pour gérer les profils de pare-feu Windows.

aws:executeAutomation - Appels -ManageWindowsService pour gérer le service RDP.AWSSupport

aws:executeAutomation - Appels -ManageRDPSettings pour gérer les paramètres RDP.AWSSupport

4. (Correction hors connexion) Si l'instance n'est pas une instance gérée :
   a. aws:assertAwsResourceProperty : imposer AllowOffline = True
   b. aws:assertAwsResourceProperty : action d'assertion =FixAll
   c. aws:assertAwsResourceProperty : impose la valeur deSubnetId

      (Utilisez le sous-réseau de l'instance fournie) Si SubnetId a pour valeur SELECTED_INSTANCE_SUBNET

      aws:executeAwsApi : récupérez le sous-réseau de l'instance actuelle.

      aws:executeAutomation : Exécutez AWSSupport -ExecuteEC2Rescue avec le sous-réseau de l'instance fournie.

   d. (Utilisez le sous-réseau personnalisé fourni) Si nSubnetId'est pas SELECTED_INSTANCE_SUBNET

      aws:executeAutomation : Exécutez AWSSupport -ExecuteEC2Rescue avec SubnetId la valeur fournie.

Résultats

.SortiemanageFirewallProfiles

.SortiemanageRDPServiceSettings

.SortiemanageRDPSettings

.SortiecheckFirewallProfiles

.SortiecheckRDPServiceSettings

.SortiecheckRDPSettings

.SortiedisableFirewallProfiles

.SortierestoreDefaultRDPServiceSettings

.SortierestoreDefaultRDPSettings

.SortietroubleshootRDPOffline

troubleshootRDPOfflineWithSubnetId.Output

AWSSupport-TroubleshootS3PublicRead

Description

Le document AWSSupport Automation -TroubleshootS3PublicRead diagnostique les problèmes de lecture des objets à partir du compartiment Amazon Simple Storage Service (Amazon S3) public que vous spécifiez dans le `S3BucketName` paramètre. Un sous-ensemble de paramètres est également analysé pour les objets du compartiment S3.

Exécution de cette automatisation (console)

Restrictions

- Cette automatisation ne vérifie pas les points d'accès qui autorisent l'accès public aux objets.
- Cette automatisation n'évalue pas les clés de condition dans la stratégie de compartiment S3.
- Si vous utilisez AWS Organizations , cette automatisation n'évalue pas les stratégies de contrôle de service pour confirmer que l'accès à Amazon S3 est autorisé.

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- CloudWatchLogGroupName

  Type: String (chaîne)

  Description : (Facultatif) Groupe de Amazon CloudWatch Logs journaux dans lequel vous souhaitez envoyer la sortie d'Automation. Si aucun groupe de journaux correspondant à la valeur que vous spécifiez n'est trouvé, Automation crée un groupe de journaux à l'aide de cette valeur de paramètre. La période de conservation pour le groupe de journaux créé par cette automatisation est de 14 jours.
- CloudWatchLogStreamName

  Type: String (chaîne)

Description : (Facultatif) Flux CloudWatch Logs de journaux dans lequel vous souhaitez envoyer la sortie d'Automation. Si aucun flux de journaux correspondant à la valeur que vous spécifiez n'est trouvé, Automation crée un flux de journaux à l'aide de cette valeur de paramètre. Si vous ne spécifiez pas de valeur pour ce paramètre, l'automatisation utilisera le `ExecutionId` comme nom du flux de journaux.

- HttpGet

  Type: Booléen

  Valeurs valides : true | false

  Valeur par défaut : true

  Description : (Facultatif) Si ce paramètre est défini sur `true` , Automation effectue une demande HTTP partielle sur les objets dans le `S3BucketName` que vous spécifiez. Seul le premier octet de l'objet est renvoyé à l'aide de l'en-tête HTTP Range.

- IgnoreBlockPublicAccess

  Type: Booléen

  Valeurs valides : true | false

  Par défaut : faux

  Description : (Facultatif) Si ce paramètre est défini sur `true` , l'automatisation ignore les paramètres de blocage d'accès public du compartiment S3 que vous spécifiez dans le `S3BucketName` paramètre. Il n'est pas recommandé de modifier la valeur par défaut de ce paramètre.

- MaxObjects

  Type: Integer

  Valeurs autorisées : 1 à 25

  Par défaut : 5

  Description : (Facultatif) Nombre d'objets à analyser dans le compartiment S3 que vous spécifiez dans le `S3BucketName` paramètre.

- S3BucketName

  Type: String (chaîne)

  Description : (Obligatoire) Nom du compartiment S3 à dépanner.

- S3PrefixName

  Type: String (chaîne)

  Description : (Facultatif) Préfixe de nom de clé des objets que vous souhaitez analyser dans votre compartiment S3. Pour plus d'informations, consultez Clés d'objet dans le Amazon Simple Storage Service Manuel du développeur .

- StartAfter

  Type: String (chaîne)

  Description : (Facultatif) Nom de la clé d'objet dans laquelle vous souhaitez que l'automatisation commence à analyser les objets de votre compartiment S3.

- ResourcePartition

  Type: String (chaîne)

Valeurs autorisées : aws | aws-us-gov | aws-cn

Par défaut: aws

Description : (Obligatoire) Partition dans laquelle se trouve votre compartiment S3.

- Détaillée

  Type: Booléen

  Valeurs valides : true | false

  Par défaut : faux

  Description : (Facultatif) Pour renvoyer des informations plus détaillées pendant l'automatisation, définissez ce paramètre sur `true` . Seuls les messages d'avertissement et d'erreur sont renvoyés si le paramètre est défini sur `false` .

Autorisations IAM nécessaires

`AutomationAssumeRole` nécessite les actions suivantes pour exécuter correctement le document Automation.

Les `logs:CreateLogGroup` autorisations `logs:CreateLogStream` , et ne sont requises que si vous souhaitez que Automation envoie les données de journaux à `logs:PutLogEvents` .CloudWatch Logs

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "iam:SimulateCustomPolicy",
                "iam:GetContextKeysForCustomPolicy",
                "s3:ListAllMyBuckets",
                "logs:CreateLogGroup",
                "logs:CreateLogStream",
                "logs:PutLogEvents",
                "logs:PutRetentionPolicy",
                "s3:GetAccountPublicAccessBlock"
            ],
            "Resource": "*",
            "Effect": "Allow"
        },
        {
            "Action": [
                "s3:GetObject",
                "s3:GetObjectAcl",
                "s3:GetObjectTagging"
            ],
            "Resource": "arn:aws:s3:::awsexamplebucket1/*",
            "Effect": "Allow"
        },
        {
            "Action": [
                "s3:ListBucket",
                "s3:GetBucketLocation",
                "s3:GetBucketPublicAccessBlock",
                "s3:GetBucketRequestPayment",
                "s3:GetBucketPolicyStatus",
                "s3:GetBucketPolicy",
                "s3:GetBucketAcl"
            ],
```

```
            "Resource": "arn:aws:s3:::awsexamplebucket1",
            "Effect": "Allow"
        }
    ]
}
```

Étapes de document

- aws:assertAwsResourceProperty - Confirme que le compartiment S3 existe et est accessible.
- aws:executeScript : renvoie l'emplacement du compartiment S3 et votre ID d'utilisateur canonique.
- aws:executeScript : Renvoie les paramètres de blocage de l'accès public pour votre compte et le compartiment S3.
- aws:assertAwsResourceProperty - Confirme que le payeur du compartiment S3 est défini sur `BucketOwner` . Si `Requester Pays` est activé sur le compartiment S3, l'automatisation se termine.
- aws:executeScript : renvoie le statut de la stratégie de compartiment S3 et détermine si elle est considérée comme publique. Pour plus d'informations sur les compartiments S3 publics, consultez Signification de « public » dans le Amazon Simple Storage Service Manuel du développeur .
- aws:executeAwsApi : renvoie la stratégie de compartiment S3.
- aws:executeAwsApi - Renvoie toutes les clés de contexte trouvées dans la stratégie de compartiment S3.
- aws:assertAwsResourceProperty - Confirme s'il existe un refus explicite dans la stratégie de compartiment S3 pour l`GetObject`'action d'API.
- aws:executeAwsApi : renvoie la liste de contrôle d'accès (ACL) pour le compartiment S3.
- aws:executeScript - Crée un groupe de CloudWatch Logs journaux et un flux de journaux si vous spécifiez une valeur pour le `CloudWatchLogGroupName` paramètre.
- aws:executeScript : en fonction des valeurs que vous spécifiez dans les paramètres d'entrée du document Automation, évalue si l'un des paramètres de compartiment S3 collectés au cours de l'automatisation empêche l'accès public aux objets. Ce script exécute les fonctions suivantes :
  - Permet d'évaluer les paramètres de bloc d'accès public
  - Renvoie les objets de votre compartiment S3 en fonction des valeurs que vous spécifiez dans les `MaxObjects` paramètres `S3PrefixName` , et .`StartAfter`
  - Renvoie la stratégie de compartiment S3 pour simuler une IAM stratégie personnalisée pour les objets renvoyés à partir de votre compartiment S3.
  - Effectue une demande HTTP partielle sur les objets renvoyés si le `HttpGet` paramètre est défini sur `true` . Seul le premier octet de l'objet est renvoyé à l'aide de l'en-tête HTTP Range.
  - Vérifie le nom de clé de l'objet renvoyé pour confirmer s'il se termine par un ou deux points. Les noms des clés d'objet qui se terminent par des points ne peuvent pas être téléchargés à partir de la Amazon S3 console .
  - Vérifie si le propriétaire de l'objet renvoyé correspond au propriétaire du compartiment S3.
  - Vérifie si la liste ACL de l'objet accorde `READ` des `FULL_CONTROL` autorisations ou aux utilisateurs anonymes.
  - Renvoie les balises associées à l'objet.
  - Utilise la IAM stratégie simulée pour confirmer s'il existe un refus explicite pour cet objet dans la stratégie de compartiment S3 pour l'action `GetObject` d'API.
  - Renvoie les métadonnées de l'objet pour confirmer que la classe de stockage est prise en charge.
  - Vérifie les paramètres de chiffrement côté serveur de l'objet pour confirmer si l'objet est chiffré à l'aide d'une clé principale client AWS Key Management Service (AWS KMS) .

Résultats

AnalyzeObjects.compartiment

AnalyzeObjectsobjet

AWSSupport-DépannageSSH

Description

Le document d'automatisation AWSSupport -TroubleshootSSH installe l'EC2Rescueoutil Amazon pour Linux, puis utilise l'EC2Rescueoutil pour vérifier ou tenter de résoudre les problèmes courants qui empêchent une connexion à distance à la machine Linux via SSH. Le cas échéant, les modifications peuvent être appliquées hors connexion par l'arrêt et le redémarrage de l'instance, si l'utilisateur autorise explicitement la correction hors connexion. Par défaut, le document fonctionne en mode lecture seule.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Linux

Paramètres

- Action

  Type: String (chaîne)

  Valeurs valides : CheckAll|FixAll

  Par défaut : CheckAll

  Description : (Obligatoire) Spécifiez si vous souhaitez vérifier les problèmes sans les corriger ou vérifier et corriger automatiquement les problèmes détectés.
- AllowOffline

  Type: String (chaîne)

  Valeurs valides : Vrai | Faux

  Par défaut : Faux

  Description : (Facultatif) Correctif uniquement - Définissez ce paramètre sur true si vous autorisez la correction SSH hors connexion dans le cas où le dépannage en ligne échouerait ou dans le cas où l'instance fournie ne serait pas une instance gérée. Remarque: Pour la correction hors connexion, SSM Automation arrête l'instance et crée une AMI avant d'essayer toute autre opération.
- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- InstanceId

  Type: String (chaîne)

Description : (Obligatoire) ID de votre instance EC2 pour Linux.

- S3BucketName

  Type: String (chaîne)

  Description : (Facultatif) Offline only - Nom du compartiment S3 de votre compte dans lequel vous souhaitez charger les journaux de dépannage. Assurez-vous que la stratégie de compartiment n'accorde pas des autorisations en lecture/écriture superflues pour les tiers qui n'ont pas besoin d'accéder aux journaux collectés.

- SubnetId

  Type: String (chaîne)

  Par défaut : SelectedInstanceSubnet

  Description : (Facultatif) Offline only - ID de sous-réseau de l'EC2Rescueinstance utilisée pour effectuer le dépannage hors connexion. Si aucun ID de sous-réseau n'est spécifié, AWS Systems Manager Automation crée un VPC.

  > **Important**
  >
  > Le sous-réseau doit être dans la même zone de disponibilité qu' InstanceId , et doit autoriser l'accès aux points de terminaison SSM.

Autorisations IAM nécessaires

`AutomationAssumeRole` nécessite les actions suivantes pour exécuter correctement le document Automation.

Il est recommandé que l'instance EC2 recevant la commande dispose d'un rôle IAM avec la stratégie gérée AmazonSSMManagedInstanceCore Amazon attachée. Pour la correction en ligne, l'utilisateur doit avoir au moins ssm:DescribeInstanceInformation, ssm:ExecuteAutomation et ssm:SendCommand pour exécuter l'instance d'Automation et envoyer la commande à l'instance, plus ssm:GetAutomationExecution pour pouvoir lire la sortie d'Automation. Pour la correction hors connexion, l'utilisateur doit avoir au moins ssm:DescribeInstanceInformation, ssm:ExecuteAutomation, ec2:DescribeInstances, plus ssm:GetAutomationExecution pour pouvoir lire la sortie de l'automatisation. AWSSupport-TroubleshootSSH calls AWSSupport -ExecuteEC2Rescue to perform the offline remediation - please review the permissions for AWSSupport -ExecuteEC2Rescue to sure you can run the automation.

Étapes de document

1. aws:assertAwsResourceProperty : vérifiez s'il s'agit d'une instance gérée.
   a. (Correction en ligne) Si l'instance est une instance gérée :
      i. aws:configurePackage : installez EC2Rescue pour Linux via AWS-ConfigureAWSPackage.
      ii. aws:runCommand : Exécutez le script bash à exécuter EC2Rescue pour Linux.
   b. (Correction hors connexion) Si l'instance n'est pas une instance gérée :
      i. aws:assertAwsResourceProperty : imposer AllowOffline = True
      ii. aws:assertAwsResourceProperty : action d'assertion =FixAll
      iii. aws:assertAwsResourceProperty : impose la valeur deSubnetId
      iv. (Utilisez le sous-réseau de l'instance fournie) Si SubnetId est SelectedInstanceSubnet sur us aws:executeAutomation pour exécuter AWSSupport -ExecuteEC2Rescue avec le sous-réseau de l'instance fournie.
      v. (Utilisez le sous-réseau personnalisé fourni) Si n'SubnetIdutilise pas aws:executeAutomation pour exécuter SelectedInstanceSubnet -ExecuteEC2Rescue avec AWSSupport la valeur fournie.SubnetId

Résultats

SortietroubleshootSSH

.SortietroubleshootSSHOffline

troubleshootSSHOfflineWithSubnetId.Output

AWS-UpdateCloudFormationStack

Description

Mettez à jour une pile AWS CloudFormation à l'aide d'un modèle AWS CloudFormation stocké dans un compartiment Amazon S3.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- LambdaAssumeRole

  Type: String (chaîne)

  Description : (Obligatoire) ARN du rôle assumé parLambda
- StackNameOrId

  Type: String (chaîne)

  Description : (Obligatoire) Nom ou ID unique de la AWS CloudFormation pile à mettre à jour
- TemplateUrl

  Type: String (chaîne)

  Description : (Obligatoire) Emplacement du compartiment S3 qui contient le CloudFormation modèle mis à jour (par exemple, https://s3.amazonaws.com/doc-example-bucket/updated.template)

AWS-UpdateLinuxAmi

Description

Mettez à jour une Amazon Machine Image (AMI) Linux avec des packages de distribution et des logiciels Amazon.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- ExcludePackages

  Type: String (chaîne)

  Par défaut: Aucun

  Description : (Facultatif) Noms des packages spécifiques à exclure des mises à jour, sous toutes les conditions. Avec la valeur par défaut ("none"), aucun package n'est exclu.
- IamInstanceProfileName

  Type: String (chaîne)

  Par défaut : ManagedInstanceProfile

  Description : (Obligatoire) Profil d'instance qui permet Systems Manager à de gérer l'instance.
- IncludePackages

  Type: String (chaîne)

  Par défaut : all

  Description : (Facultatif) Mettre à jour uniquement ces packages nommés. Avec la valeur par défaut ("all"), toutes les mises à jour disponibles sont appliquées.
- InstanceType

  Type: String (chaîne)

  Par défaut : t2.micro

  Description : (Facultatif) Type d'instance à lancer en tant qu'hôte d'espace de travail. Les types d'instances varient selon la région.
- PostUpdateScript

  Type: String (chaîne)

Par défaut: Aucun

Description : (Facultatif) URL d'un script à exécuter après l'application des mises à jour de package. La valeur par défaut ("none") implique la non-exécution d'un script.

- PreUpdateScript

  Type: String (chaîne)

  Par défaut: Aucun

  Description : (Facultatif) URL d'un script à exécuter avant l'application des mises à jour. La valeur par défaut ("none") implique la non-exécution d'un script.

- SourceAmiId

  Type: String (chaîne)

  Description : (Obligatoire) ID de l'Amazon Machine Image source.

- TargetAmiName

  Type: String (chaîne)

  Par défaut : UpdateLinuxAmi_from_{{SourceAmiId}}_on_{{global:DATE_TIME}}

  Description : (Facultatif) Nom de la nouvelle AMI qui sera créée. La valeur par défaut est une chaîne générée par le système qui inclut l'ID de l'AMI source, et les date et heure de création.

AWS-UpdateWindowsAmi

Description

Mettez à jour une Amazon Machine Image (AMI) Microsoft Windows. Par défaut, ce document installe toutes les mises à jour Windows, ainsi que les logiciel et les pilotes Amazon. Ensuite, il exécute Sysprep pour créer une AMI. Prend en charge Windows Server 2008 R2 ou version ultérieure.

> **Important**
>
> Si vos instances se connectent à AWS Systems Manager à l'aide de points de terminaison d'un VPC, ce document échouera à moins qu'il ne soit utilisé dans la région us-east-1.

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AutomationAssumeRole

  Type : chaîne

Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.

- Catégories

  Type: String (chaîne)

  Description : (Facultatif) Spécifiez une ou plusieurs catégories de mise à jour. Vous pouvez filtrer les catégories en utilisant des valeurs séparées par une virgule. Options: Applications, connecteurs, CriticalUpdates , DefinitionUpdates , DeveloperKits , pilotes, FeaturePacks , conseils, Microsoft, SecurityUpdates , outils, mises à jour.ServicePacksUpdateRollups Les formats valides incluent une seule entrée, par exemple : CriticalUpdates. Ou vous pouvez spécifier une liste séparée par des virgules : CriticalUpdates, SecurityUpdates. REMARQUE : Les virgules ne peuvent pas être entourées d'espaces.

- ExcludeKbs

  Type: String (chaîne)

  Description : (Facultatif) Spécifiez un ou plusieurs articles Microsoft Knowledge Base (KB) IDs à exclure. Vous pouvez en exclure plusieurs IDs à l'aide de valeurs séparées par des virgules. Formats valides : KB9876543 ou 9876543.

- IamInstanceProfileName

  Type: String (chaîne)

  Par défaut : ManagedInstanceProfile

  Description : (Obligatoire) Nom du rôle qui permet Systems Manager à de gérer l'instance.

- IncludeKbs

  Type: String (chaîne)

  Description : (Facultatif) Spécifiez un ou plusieurs articles Microsoft Knowledge Base (KB) IDs à inclure. Vous pouvez en installer plusieurs IDs à l'aide de valeurs séparées par des virgules. Formats valides : KB9876543 ou 9876543.

- InstanceType

  Type: String (chaîne)

  Par défaut : t2.medium

  Description : (Facultatif) Type d'instance à lancer en tant qu'hôte d'espace de travail. Les types d'instances varient selon la région. La valeur par défaut est t2.medium.

- PostUpdateScript

  Type: String (chaîne)

  Description : (Facultatif) Script fourni sous forme de chaîne. Il est exécuté après l'installation des mises à jour du système d'exploitation.

- PreUpdateScript

  Type: String (chaîne)

  Description : (Facultatif) Script fourni sous forme de chaîne. Il est exécuté avant l'installation des mises à jour du système d'exploitation.

- PublishedDateAfter

  Type: String (chaîne)

Description : (Facultatif) Spécifiez la date après laquelle les mises à jour doivent être publiées. Par exemple, si 01/01/2017 est spécifié, toutes les mises à jour qui ont été trouvées pendant la recherche des mises à jour Windows qui ont été publiées après le 01/01/2017 sont renvoyées.

- PublishedDateBefore

  Type: String (chaîne)

  Description : (Facultatif) Spécifiez la date avant laquelle les mises à jour doivent être publiées. Par exemple, si 01/01/2017 est spécifié, toutes les mises à jour qui ont été trouvées pendant la recherche des mises à jour Windows qui ont été publiées avant le 01/01/2017 sont renvoyées.
- PublishedDaysOld

  Type: String (chaîne)

  Description : (Facultatif) Spécifiez le nombre de jours pendant lesquels les mises à jour doivent remonter à partir de la date de publication. Par exemple, si 10 est spécifié, toutes les mises à jour qui ont été trouvées pendant la recherche des mises à jour Windows il y a 10 jours ou plus sont renvoyées.
- SeverityLevels

  Type: String (chaîne)

  Description : (Facultatif) Spécifiez un ou plusieurs niveaux de sécurité MSRC associés à une mise à jour. Vous pouvez filtrer les niveaux de sécurité en utilisant des valeurs séparées par une virgule. Par défaut, les correctifs pour tous les niveaux de sécurité sont sélectionnés. Si cette valeur est fournie, la liste de mise à jour est filtrée en conséquence. Options: Critique, Important, Faible, Modéré ou Non précisé. Les formats valides incluent une seule entrée, par exemple : Critique. Vous pouvez également spécifier une liste séparée par des virgules : Critique, Important, Faible.
- SourceAmiId

  Type: String (chaîne)

  Description : (Obligatoire) ID de l'Amazon Machine Image source.
- SubnetId

  Type: String (chaîne)

  Description : (Facultatif) Spécifiez le SubnetId si vous souhaitez lancer dans un sous-réseau spécifique.
- TargetAmiName

  Type: String (chaîne)

  Par défaut : UpdateWindowsAmi_from_{{SourceAmiId}}_on_{{global:DATE_TIME}}

  Description : (Facultatif) Nom de la nouvelle AMI qui sera créée. La valeur par défaut est une chaîne générée par le système qui inclut l'ID de l'AMI source, et les date et heure de création.

AWSSupport-UpgradeWindowsAWSDrivers (Mettre à niveauWindowsAWSDrivers)

Description

-UpgradeWindowsAWSDrivers met à niveau ou répare les AWSSupport pilotes de stockage et de réseau sur l'instance EC2 spécifiée.AWS Le document tente d'installer les dernières versions des pilotes AWS en ligne en appelant l'Agent SSM. Si l'Agent SSM ne peut pas être contacté, le document peut effectuer une installation hors ligne des pilotes AWS si elle est explicitement demandée. Remarque: La mise à niveau en ligne et hors connexion créera une AMI avant d'essayer toute opération, qui sera conservée une fois

l'automatisation terminée. Vous devez sécuriser l'accès à l'AMI ou le supprimer. La méthode en ligne redémarre l'instance dans le cadre du processus de mise à niveau, tandis que la méthode hors connexion nécessite l'arrêt, puis le redémarrage de l'instance EC2 fournie.

> **Important**
>
> Si vos instances se connectent à AWS Systems Manager à l'aide de points de terminaison d'un VPC, ce document échouera à moins qu'il ne soit utilisé dans la région us-east-1. Ce document échoue également sur un contrôleur de domaine. Pour mettre à jour les pilotes AWS PV sur un contrôleur de domaine, veuillez consulter Mise à niveau d'un contrôleur de domaine (Mise à niveau d'AWS PV).

Exécution de cette automatisation (console)

Type de document

Automation

Propriétaire

Amazon

Plateformes

Windows, Linux

Paramètres

- AllowOffline

  Type: String (chaîne)

  Valeurs valides : Vrai | Faux

  Par défaut : Faux

  Description : (Facultatif) Définissez ce paramètre sur true si vous autorisez une mise à niveau des pilotes hors ligne au cas où l'installation en ligne ne pourrait pas être effectuée. Remarque: La méthode hors connexion nécessite l'arrêt, puis le démarrage de l'instance EC2. Les données stockées sur les volumes de stockage d'instance seront perdues. L'adresse IP publique change si vous n'utilisez pas une adresse IP Elastic.
- AutomationAssumeRole

  Type : chaîne

  Description : (Facultatif) Amazon Resource Name (ARN) du rôle AWS Identity and Access Management (IAM) qui autorise Systems Manager Automation à effectuer les actions en votre nom. Si aucun rôle n'est spécifié, Systems Manager Automation utilise les autorisations de l'utilisateur qui exécute ce document.
- ForceUpgrade

  Type: String (chaîne)

  Valeurs valides : Vrai | Faux

  Par défaut : Faux

  Description : (Facultatif) Offline only - Définissez cette valeur sur true si vous autorisez la mise à niveau des pilotes hors connexion même si les pilotes les plus récents sont déjà installés sur votre instance.
- InstanceId

Type: String (chaîne)

Description : (Obligatoire) ID de votre instance EC2 pour Windows Server .

- SubnetId

  Type: String (chaîne)

  Par défaut : SelectedInstanceSubnet

  Description : (Facultatif) Offline only - ID de sous-réseau de l'EC2Rescueinstance utilisée pour effectuer la mise à niveau des pilotes hors connexion. Si aucun ID de sous-réseau n'est spécifié, Systems Manager Automation crée un VPC.

  > **Important**
  >
  > Le sous-réseau doit être dans la même zone de disponibilité qu' InstanceId et doit autoriser l'accès aux points de terminaison SSM.

Autorisations IAM nécessaires

`AutomationAssumeRole` nécessite les actions suivantes pour exécuter correctement le document Automation.

L'instance EC2 recevant la commande doit au minimum avoir un rôle IAM qui inclut les autorisations pour ssm:StartAutomationExecution et ssm:SendCommand pour exécuter l'automatisation et envoyer la commande à l'instance, plus ssm:GetAutomationExecution pour pouvoir lire la sortie d'automatisation. Vous pouvez joindre la stratégie gérée par `AmazonSSMManagedInstanceCore` Amazon à votre rôle IAM pour fournir ces autorisations. Nous vous recommandons toutefois d'utiliser le rôle IAM Automation `AmazonSSMAutomationRole` à cette fin. Pour plus d'informations, consultez la section Méthode 2 : Utiliser IAM pour configurer des rôles pour Automation (p. 310).

Si vous effectuez une mise à niveau hors connexion, veuillez consulter les autorisations requises par AWSSupport-StartEC2RescueWorkflow (p. 600).

Étapes de document

1. aws:assertAwsResourceProperty : vérifie que l'instance d'entrée est Windows.
2. aws:assertAwsResourceProperty : vérifie que l'instance d'entrée est une instance gérée. Si tel est le cas, la mise à niveau en ligne démarre. Dans le cas contraire, la mise à niveau hors connexion est évaluée.
   a. (Mise à niveau en ligne) Si l'instance d'entrée est une instance gérée :
      i. aws:createImage : crée une sauvegarde de l'AMI.
      ii. aws:createTags : balise la sauvegarde de l'AMI.
      iii. aws:runCommand : installe le pilote réseau ENA via AWS-ConfigureAWSPackage.
      iv. aws:runCommand : installe NVMe le pilote via AWS-ConfigureAWSPackage.
      v. aws:runCommand : installe le pilote réseau AWS PV via AWS-ConfigureAWSPackage.
   b. (Mise à niveau hors connexion) Si l'instance d'entrée n'est pas une instance gérée :
      i. aws:assertAwsResourceProperty : vérifie que l'AllowOfflineindicateur est défini sur True. Si tel est le cas, la mise à niveau hors connexion démarre. Dans le cas contraire, le flux de travail se termine.
      ii. aws:changeInstanceState : arrêtez l'instance source.
      iii. aws:changeInstanceState : forcez l'arrêt de l'instance source.
      iv. aws:createImage : créez une sauvegarde d'AMI de l'instance source.
      v. aws:createTags : balisez la sauvegarde d'AMI de l'instance source.
      vi. aws:executeAwsApi : activez ENA pour l'instance.

vii. aws:assertAwsResourceProperty : imposez l'ForceUpgradeindicateur.

viiiniveau forcé hors connexion) Si ForceUpgrade = True, exécutez aws:executeAutomation pour appeler AWSSupport -StartEC2RescueWorkflow avec le script de mise à niveau forcée des pilotes. Cette opération installe les pilotes quelle que soit la version actuelle qui est installée.

ix. (Mise à niveau hors connexion) Si ForceUpgrade = False, exécutez aws:executeAutomation pour appeler AWSSupport -StartEC2RescueWorkflow avec le script de mise à niveau des pilotes.

Résultats

preUpgradeBackupID d'image

preOfflineUpgradeBackupID d'image

SortieinstallAwsEnaNetworkDriverOnInstance

.SortieinstallAWSNVMeOnInstance

.SortieinstallAWSPVDriverOnInstance

.SortieupgradeDriversOffline

.SortieforceUpgradeDriversOffline

Procédures d'automatisation

Les procédures suivantes vous aident à faire vos premiers pas avec Systems Manager Automation à l'aide de documents Automation prédéfinis.

Avant de commencer, vous devez configurer les rôles et les autorisations Automation. Pour plus d'informations, consultez la section Configuration d'Automation (p. 308). Pour de plus amples informations sur la création d'un document d'Automation personnalisé, veuillez consulter Procédure : Appliquer un correctif à une AMI Windows Server (p. 640).

Procédures

- Application de correctifs aux images AMI (Amazon Machine Image) (p. 632)
- Utilisation des instances Automation en libre-service du support AWS (p. 656)
- Procédure : Utilisation des transformateurs en entrée avec Automation (p. 666)
- Procédure : Utilisation d'Automation avec Jenkins (p. 667)
- Procédure : Utilisation de Document Builder pour créer un document Automation personnalisé (p. 670)

Application de correctifs aux images AMI (Amazon Machine Image)

Cette section inclut des procédures qui décrivent comment corriger ou mettre à jour des Amazon Machine Images (AMI).

Rubriques

- Procédure : Appliquer un correctif à une AMI Linux (console) (p. 633)
- Procédure : Appliquer un correctif à une AMI Linux (AWS CLI) (p. 636)
- Procédure : Appliquer un correctif à une AMI Windows Server (p. 640)
- Procédure : Simplification de l'application de correctifs à l'AMI avec Automation, AWS Lambda et Parameter Store (p. 645)
- Procédure : Appliquer des correctifs à une AMI et mettre à jour un groupe Auto Scaling (p. 651)

Procédure : Appliquer un correctif à une AMI Linux (console)

Cette procédure Systems Manager Automation vous montre comment utiliser la console et le document Systems Manager`AWS-UpdateLinuxAmi` pour appliquer automatiquement un correctif à une AMI Linux avec les versions les plus récentes des packages que vous spécifiez. Le document `AWS-UpdateLinuxAmi` automatise aussi l'installation de packages et configurations supplémentaires propres aux sites. À l'aide de cette procédure, vous pouvez mettre à jour une grande variété de distributions Linux, y compris Ubuntu, CentOS , RHEL, SLES ou Amazon Linux AMIs . Pour obtenir la liste complète des versions Linux prises en charge, consultez Prérequis d'Gestionnaire de correctifs (p. 1048) .

Le document `AWS-UpdateLinuxAmi` vous permet d'automatiser les tâches de maintenance d'images sans devoir créer de flux de travail en JSON ou YAML. Vous pouvez utiliser le document `AWS-UpdateLinuxAmi` pour effectuer les types de tâche suivants.

- Mettez à niveau tous les packages de distribution et les logiciels Amazon sur une Amazon Machine Image (AMI) Amazon Linux, Red Hat, Ubuntu, SLES ou Cent OS Il s'agit du comportement de document par défaut.
- Installer Agent SSM sur une image existante pour activer des fonctionnalités Systems Manager comme l'exécution de commande distante à l'aide de la fonctionnalité Fonctionnalité Exécuter la commande ou l'extraction des inventaires logiciels avec Inventory.
- Installer des packages logiciels supplémentaires

Avant de commencer

Avant de commencer à travailler avec les documents Automation, configurez des rôles et, le cas échéant, EventBridge pour Automation. Pour plus d'informations, consultez la section Configuration d'Automation (p. 308). Cette procédure pas à pas nécessite également que vous spécifiez le nom d'un profil d'instance AWS Identity and Access Management (IAM). Pour plus d'informations sur la création d'un profil d'instance IAM, consultez la page Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34).

Le document `AWS-UpdateLinuxAmi` accepte les paramètres d'entrée suivants.

Paramètre	Tapez	Description
SourceAmiId	String (chaîne)	(Obligatoire) ID de l'AMI source.
IamInstanceProfileName	String (chaîne)	(Obligatoire) Nom du rôle de profil d'instance IAM que vous avez créé dans Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). Le rôle de profil d'instance autorise Automation à effectuer des actions sur vos instances, comme l'exécution de commandes, ou le démarrage ou l'arrêt de services. Le document d'Automation utilise uniquement le nom du rôle de profil d'instance. Si vous spécifiez l'ARN (Amazon Resource Name), l'exécution d'Automation échoue.
AutomationAssumeRole	String (chaîne)	(Obligatoire) Le nom du rôle de service IAM que vous avez créé dans Configuration

Paramètre	Tapez	Description
		d'Automation (p. 308). Le rôle de service (également appelé rôle de responsable) accorde à Automation l'autorisation d'assumer votre rôle IAM et d'exécuter des actions en votre nom. Par exemple, le rôle de service autorise Automation à créer une nouvelle AMI; lors de l'exécution de l'action `aws:createImage` dans un document Automation. Pour ce paramètre, l'ARN complet doit être spécifié.
TargetAmiName	String (chaîne)	(Facultatif) Nom de la nouvelle AMI une fois celle-ci créée. Le nom par défaut est une chaîne générée par le système qui inclut l'ID de l'AMI source, et les date et heure de création.
InstanceType	String (chaîne)	(Facultatif) Type d'instance à lancer en tant qu'hôte d'espace de travail. Les types d'instances varient selon la région. La type par défaut est t2.micro.
PreUpdateScript	String (chaîne)	(Facultatif) URL d'un script à exécuter avant l'application des mises à jour. La valeur par défaut (\"none\") consiste à ne pas exécuter de script.
PostUpdateScript	String (chaîne)	(Facultatif) URL d'un script à exécuter après l'application des mises à jour de package. La valeur par défaut (\"none\") consiste à ne pas exécuter de script.
IncludePackages	String (chaîne)	(Facultatif) Mettre à jour uniquement ces packages nommés. Par défaut (\"all \"), toutes les mises à jour disponibles sont appliquées.
ExcludePackages	String (chaîne)	(Facultatif) Noms des packages spécifiques à exclure des mises à jour, sous toutes les conditions. Par défaut (\"none\"), aucun package n'est exclu.

Étapes d'Automation

Le document `AWS-UpdateLinuxAmi` inclut les étapes d'Automation suivantes par défaut.

Étape 1 : launchInstance(action aws:runInstances)

Cette étape lance une instance à l'aide de données utilisateur (userdata) Amazon EC2 et d'un rôle de profil d'instance IAM. Userdata installe l'agent SSM approprié selon le système d'exploitation. L'installation de Agent SSM vous permet d'utiliser des fonctionnalités Systems Manager comme Fonctionnalité Exécuter la commande, State Manager et Inventory.

Étape 2 : updateOSSoftware(aws:runCommand action)

Cette étape exécute les commandes suivantes sur l'instance lancée :

- Téléchargement d'un script de mise à jour depuis Amazon S3.
- Exécution d'un script de pré-mise à jour facultatif.
- Mise à jour des packages de distribution et des logiciels Amazon.
- Exécution d'un script de post-mise à jour facultatif.

Le journal d'exécution est stocké dans le dossier /tmp pour que l'utilisateur puisse le consulter ultérieurement.

Si vous souhaitez mettre à niveau un ensemble de packages spécifique, vous pouvez fournir la liste à l'aide du paramètre `IncludePackages`. Dans ce cas, le système tente de mettre à jour seulement ces packages et leurs dépendances. Aucune autre mise à jour n'est exécutée. Par défaut, quand aucun package include n'est spécifié, le programme met à jour tous les packages disponibles.

Si vous souhaitez exclure de la mise à niveau un ensemble de packages spécifique, vous pouvez fournir la liste à l'aide du paramètre `ExcludePackages`. Dans ce cas, ces packages restent à leur version actuelle, quelles que soient les autres options spécifiées. Par défaut, quand aucun package exclude n'est spécifié, aucun package n'est exclu.

Étape 3 : stopInstance(action aws:changeInstanceState)

Cette étape arrête toutes les instances mises à jour.

Étape 4 : createImage(action aws:createImage)

Cette étape crée une nouvelle AMI avec un nom descriptif qui la relie à l'ID source et l'heure de création. Exemple : AMI générée par EC2 Automation sur {{global:DATE_TIME}} à partir de {{SourceAmiId}} où DATE_TIME et SourceID représentent des variables Automation.

Étape 5 : terminateInstance(action aws:changeInstanceState)

Cette étape nettoie l'exécution en mettant hors service l'instance en cours d'exécution.

Output

L'exécution renvoie l'ID de la nouvelle AMI en tant que sortie.

> **Note**
>
> Par défaut, lorsqu'Automation exécute le document AWS-`AWS-UpdateLinuxAmi`, le système crée une instance temporaire dans le VPC par défaut (172.30.0.0/16). Si vous avez supprimé le VPC par défaut, vous recevrez l'erreur suivante :
> `VPC not defined 400`
> Pour régler ce problème, vous devez réaliser une copie du document `AWS-UpdateLinuxAmi` et spécifier un ID de sous-réseau. Pour plus d'informations, consultez la section VPC non défini 400 (p. 677).

Pour créer une AMI à laquelle un correctif a été appliqué l'aide d'Automation (AWS Systems Manager)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Automatisation.

-ou-

Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Automation.

3. Choisissez Execute automation (Exécuter l'automatisation).
4. Dans la liste Document d'automatisation, choisissez AWS-UpdateLinuxAmi.
5. Dans la section Document details (Détails du document), vérifiez que l'option Document version (Version du document) est définie sur Version par défaut lors de l'exécution.
6. Choisissez Nex t(Suivant).
7. Dans la section Execution mode (Mode d'exécution), choisissez Simple Execution (Exécution simple).
8. Dans la section Paramètres d'entrée, entrez les informations que vous avez collectées dans la section Avant de commencer.
9. Sélectionnez Exécuter. La console affiche le statut de l'exécution d'automatisation.

Lorsque le flux de travail est terminé, lancez une instance test à partir de l'AMI mise à jour pour vérifier les modifications.

> **Note**
>
> Si une étape du flux de travail échoue, les informations concernant l'échec sont répertoriées dans la page Automation Executions. Le flux de travail est conçu pour mettre fin à l'instance temporaire une fois que toutes les tâches ont été effectuées correctement. Si une étape échoue, le système ne met pas nécessairement fin à l'instance. Donc, si une étape échoue, mettez fin à l'instance temporaire manuellement.

Procédure : Appliquer un correctif à une AMI Linux (AWS CLI)

Cette procédure Systems Manager Automation vous montre comment utiliser l'AWS CLI et le document Systems Manager `AWS-UpdateLinuxAmi` pour appliquer automatiquement un correctif à une AMI Linux avec les versions les plus récentes des packages que vous spécifiez. Le document `AWS-UpdateLinuxAmi` automatise aussi l'installation de packages et configurations supplémentaires propres aux sites. A l'aide de cette procédure, vous pouvez mettre à jour une grande variété de distributions Linux, y compris Ubuntu, CentOS , RHEL, SLES ou Amazon Linux AMIs . Pour obtenir la liste complète des versions Linux prises en charge, consultez Prérequis d'Gestionnaire de correctifs (p. 1048) .

Le document `AWS-UpdateLinuxAmi` vous permet d'automatiser les tâches de maintenance d'images sans devoir créer de flux de travail en JSON ou YAML. Vous pouvez utiliser le document `AWS-UpdateLinuxAmi` pour effectuer les types de tâche suivants.

- Mettez à niveau tous les packages de distribution et les logiciels Amazon sur une Amazon Machine Image (AMI) Amazon Linux, Red Hat, Ubuntu, SLES ou Cent OS Il s'agit du comportement de document par défaut.
- Installer Agent SSM sur une image existante pour activer des fonctionnalités Systems Manager comme l'exécution de commande distante à l'aide de la fonctionnalité Fonctionnalité Exécuter la commande ou l'extraction des inventaires logiciels avec Inventory.
- Installer des packages logiciels supplémentaires

Avant de commencer

Avant de commencer à travailler avec les documents Automation, configurez des rôles et, le cas échéant, EventBridge pour Automation. Pour plus d'informations, consultez la section Configuration d'Automation (p. 308). Cette procédure pas à pas nécessite également que vous spécifiez le nom d'un profil d'instance AWS Identity and Access Management (IAM). Pour plus d'informations sur la création d'un profil d'instance IAM, consultez la page Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34).

Le document `AWS-UpdateLinuxAmi` accepte les paramètres d'entrée suivants.

Paramètre	Tapez	Description
SourceAmiId	String (chaîne)	(Obligatoire) ID de l'AMI source. Vous pouvez référencer automatiquement le dernier ID d'une AMI EC2 pour Linux à l'aide d'un paramètre Systems Manager Parameter Store public. Pour plus d'informations, consultez Requête portant sur la dernière AMI Amazon Linux IDs à l'aide AWS Systems Manager du Parameter Store.
IamInstanceProfileName	String (chaîne)	(Obligatoire) Nom du rôle de profil d'instance IAM que vous avez créé dans Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). Le rôle de profil d'instance autorise Automation à effectuer des actions sur vos instances, comme l'exécution de commandes, ou le démarrage ou l'arrêt de services. Le document d'Automation utilise uniquement le nom du rôle de profil d'instance. Si vous spécifiez l'ARN (Amazon Resource Name), l'exécution d'Automation échoue.
AutomationAssumeRole	String (chaîne)	(Obligatoire) Le nom du rôle de service IAM que vous avez créé dans Configuration d'Automation (p. 308). Le rôle de service (également appelé rôle de responsable) accorde à Automation l'autorisation d'assumer votre rôle IAM et d'exécuter des actions en votre nom. Par exemple, le rôle de service autorise Automation à créer une nouvelle AMI; lors de l'exécution de l'action `aws:createImage` dans un document Automation. Pour ce paramètre, l'ARN complet doit être spécifié.
TargetAmiName	String (chaîne)	(Facultatif) Nom de la nouvelle AMI une fois celle-ci créée. Le nom par défaut est une chaîne générée par le système qui inclut l'ID de l'AMI source, et les date et heure de création.

Paramètre	Tapez	Description
InstanceType	String (chaîne)	(Facultatif) Type d'instance à lancer en tant qu'hôte d'espace de travail. Les types d'instances varient selon la région. La type par défaut est t2.micro.
PreUpdateScript	String (chaîne)	(Facultatif) URL d'un script à exécuter avant l'application des mises à jour. La valeur par défaut (\"none\") consiste à ne pas exécuter de script.
PostUpdateScript	String (chaîne)	(Facultatif) URL d'un script à exécuter après l'application des mises à jour de package. La valeur par défaut (\"none\") consiste à ne pas exécuter de script.
IncludePackages	String (chaîne)	(Facultatif) Mettre à jour uniquement ces packages nommés. Par défaut (\"all \"), toutes les mises à jour disponibles sont appliquées.
ExcludePackages	String (chaîne)	(Facultatif) Noms des packages spécifiques à exclure des mises à jour, sous toutes les conditions. Par défaut (\"none\"), aucun package n'est exclu.

Étapes d'Automation

Le document `AWS-UpdateLinuxAmi` inclut les étapes d'Automation suivantes par défaut.

Étape 1 : launchInstance(action aws:runInstances)

Cette étape lance une instance à l'aide de données utilisateur (userdata) Amazon EC2 et d'un rôle de profil d'instance IAM. Userdata installe l'agent SSM approprié selon le système d'exploitation. L'installation de Agent SSM vous permet d'utiliser des fonctionnalités Systems Manager comme Fonctionnalité Exécuter la commande, State Manager et Inventory.

Étape 2 : updateOSSoftware(aws:runCommand action)

Cette étape exécute les commandes suivantes sur l'instance lancée :

- Téléchargement d'un script de mise à jour depuis Amazon S3.
- Exécution d'un script de pré-mise à jour facultatif.
- Mise à jour des packages de distribution et des logiciels Amazon.
- Exécution d'un script de post-mise à jour facultatif.

Le journal d'exécution est stocké dans le dossier /tmp pour que l'utilisateur puisse le consulter ultérieurement.

Si vous souhaitez mettre à niveau un ensemble de packages spécifique, vous pouvez fournir la liste à l'aide du paramètre `IncludePackages`. Dans ce cas, le système tente de mettre à jour seulement

ces packages et leurs dépendances. Aucune autre mise à jour n'est exécutée. Par défaut, quand aucun package include n'est spécifié, le programme met à jour tous les packages disponibles.

Si vous souhaitez exclure de la mise à niveau un ensemble de packages spécifique, vous pouvez fournir la liste à l'aide du paramètre `ExcludePackages`. Dans ce cas, ces packages restent à leur version actuelle, quelles que soient les autres options spécifiées. Par défaut, quand aucun package exclude n'est spécifié, aucun package n'est exclu.

Étape 3 : stopInstance(action aws:changeInstanceState)

Cette étape arrête toutes les instances mises à jour.

Étape 4 : createImage(action aws:createImage)

Cette étape crée une nouvelle AMI avec un nom descriptif qui la relie à l'ID source et l'heure de création. Exemple : « AMI Generated by EC2 Automation on {{global:DATE_TIME}} from {{SourceAmiId}} » où DATE_TIME et SourceID représentent des variables Automation.

Étape 5 : terminateInstance(action aws:changeInstanceState)

Cette étape nettoie l'exécution en mettant hors service l'instance en cours d'exécution.

Output

L'exécution renvoie l'ID de la nouvelle AMI en tant que sortie.

Note

Par défaut, lorsqu'Automation exécute le document AWS-`AWS-UpdateLinuxAmi`, le système crée une instance temporaire dans le VPC par défaut (172.30.0.0/16). Si vous avez supprimé le VPC par défaut, vous recevrez l'erreur suivante :
`VPC not defined 400`
Pour régler ce problème, vous devez réaliser une copie du document `AWS-UpdateLinuxAmi` et spécifier un ID de sous-réseau. Pour plus d'informations, consultez la section VPC non défini 400 (p. 677).

Pour créer une AMI corrigée à l'aide de l'automatisation

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Exécutez la commande suivante pour exécuter le document `AWS-UpdateLinuxAmi` et exécuter le flux de travail Automation. Dans la section Paramètres, spécifiez un ID de source d'AMI, un rôle de profil d'instance EC2 et votre rôle de service Automation.

```
aws ssm start-automation-execution \
    --document-name "AWS-UpdateLinuxAmi" \
    --parameters \
    SourceAmiId=ami-0080e4c5bc078760e, \
    IamInstanceProfileName=ManagedInstanceRole, \
    AutomationAssumeRole='arn:aws:iam::{{global:ACCOUNT_ID}}:role/
AutomationServiceRole'"
```

   La commande renvoie un ID d'exécution. Copiez cet ID dans le Presse-papiers. Vous allez utiliser cet ID pour afficher l'état du flux de travail.

```
{
    "AutomationExecutionId": "ID"
```

```
}
```

3. Pour afficher l'exécution de flux de travail à l'aide de l'AWS CLI, exécutez la commande suivante :

```
aws ssm describe-automation-executions
```

4. Pour afficher les détails de la progression de l'exécution, exécutez la commande suivante.

```
aws ssm get-automation-execution --automation-execution-id ID
```

Le processus de mise à jour peut prendre 30 minutes ou plus.

> **Note**
>
> Vous pouvez aussi surveiller l'état du flux de travail dans la console. Dans la liste d'exécutions, choisissez l'exécution que vous venez juste d'exécuter, puis choisissez l'onglet Steps. Cet onglet affiche l'état des actions de flux de travail.

Lorsque le flux de travail est terminé, lancez une instance test à partir de l'AMI mise à jour pour vérifier les modifications.

> **Note**
>
> Si une étape du flux de travail échoue, les informations concernant l'échec sont répertoriées dans la page Automation Executions. Le flux de travail est conçu pour mettre fin à l'instance temporaire une fois que toutes les tâches ont été effectuées correctement. Si une étape échoue, le système ne met pas nécessairement fin à l'instance. Donc, si une étape échoue, mettez fin à l'instance temporaire manuellement.

Procédure : Appliquer un correctif à une AMI Windows Server

Le `AWS-UpdateWindowsAmi` document vous permet d'automatiser les tâches de maintenance d'images sur vos fenêtres Amazon AMIs sans devoir créer le flux de travail en JSON ou YAML. Ce document est pris en charge pour Windows Server 2008 R2 ou une version ultérieure. Vous pouvez utiliser le document `AWS-UpdateWindowsAmi` pour effectuer les types de tâche suivants.

- Installez toutes les mises à jour Windows et mettez à niveau les logiciels Amazon (comportement par défaut).
- Installez toutes les mises à jour Windows spécifiques et mettez à niveau les logiciels Amazon.
- Personnalisez une AMI en utilisant vos scripts.

Avant de commencer

Avant de commencer à travailler avec les documents Automation, configurez des rôles et, le cas échéant, EventBridge pour Automation. Pour plus d'informations, consultez la section Configuration d'Automation (p. 308). Cette procédure pas à pas nécessite également que vous spécifiez le nom d'un profil d'instance AWS Identity and Access Management (IAM). Pour plus d'informations sur la création d'un profil d'instance IAM, consultez la page Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34).

> **Note**
>
> Les mises à jour de Agent SSM sont généralement déployées dans différentes régions, à des heures différentes. Lorsque vous personnalisez ou mettez à jour une AMI, utilisez uniquement la source AMIs publiée pour la région dans laquelle vous travaillez. Vous aurez ainsi l'assurance de travailler avec le dernier agent Agent SSM publié pour cette région, et vous éviterez des problèmes de compatibilité.

Le document `AWS-UpdateWindowsAmi` accepte les paramètres d'entrée suivants.

Paramètre	Tapez	Description
SourceAmiId	String (chaîne)	(Obligatoire) ID de l'AMI source. Vous pouvez automatiquement référencer la dernière AMI Windows Server à l'aide d'un paramètre public du Parameter Store Systems Manager. Pour plus d'informations, consultez Interrogation pour la dernière AMI Windows IDs à l'aide d' AWS Systems Manager Parameter Store .
IamInstanceProfileName	String (chaîne)	(Obligatoire) Nom du rôle de profil d'instance IAM que vous avez créé dans Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). Le rôle de profil d'instance autorise Automation à effectuer des actions sur vos instances, comme l'exécution de commandes, ou le démarrage ou l'arrêt de services. Le document d'Automation utilise uniquement le nom du rôle de profil d'instance. Si vous spécifiez l'ARN (Amazon Resource Name), l'exécution d'Automation échoue.
AutomationAssumeRole	String (chaîne)	(Obligatoire) Le nom du rôle de service IAM que vous avez créé dans Configuration d'Automation (p. 308). Le rôle de service (également appelé rôle de responsable) accorde à Automation l'autorisation d'assumer votre rôle IAM et d'exécuter des actions en votre nom. Par exemple, le rôle de service autorise Automation à créer une nouvelle AMI; lors de l'exécution de l'action `aws:createImage` dans un document Automation. Pour ce paramètre, l'ARN complet doit être spécifié.
TargetAmiName	String (chaîne)	(Facultatif) Nom de la nouvelle AMI une fois celle-ci créée. Le nom par défaut est une chaîne générée par le système qui inclut l'ID de l'AMI source, et les date et heure de création.

Paramètre	Tapez	Description
InstanceType	String (chaîne)	(Facultatif) Type d'instance à lancer en tant qu'hôte d'espace de travail. Les types d'instances varient selon la région. La type par défaut est t2.medium.
PreUpdateScript	String (chaîne)	(Facultatif) Un script à exécuter avant la mise à jour de l'AMI. Saisissez un script dans le document d'Automation ou lors de l'exécution en tant que paramètre.
PostUpdateScript	String (chaîne)	(Facultatif) Un script à exécuter après la mise à jour de l'AMI. Saisissez un script dans le document d'Automation ou lors de l'exécution en tant que paramètre.
IncludeKbs	String (chaîne)	(Facultatif) Spécifiez un ou plusieurs articles Microsoft Knowledge Base (KB) IDs à inclure. Vous pouvez en installer plusieurs IDs à l'aide de valeurs séparées par des virgules. Formats valides : KB9876543 ou 9876543.
ExcludeKbs	String (chaîne)	(Facultatif) Spécifiez un ou plusieurs articles Microsoft Knowledge Base (KB) IDs à exclure. Vous pouvez en exclure plusieurs IDs à l'aide de valeurs séparées par des virgules. Formats valides : KB9876543 ou 9876543.
Catégories	String (chaîne)	(Facultatif) Spécifiez une ou plusieurs catégories de mise à jour. Vous pouvez filtrer les catégories en utilisant des valeurs séparées par une virgule. Options: Mise à jour critique, mise à jour de sécurité, mise à jour de définition, report de mise à jour, Service Pack, outil, mise à jour ou pilote. Parmi les formats valides figurent une seule entrée, par exemple : Mise à jour critique. Vous pouvez également spécifier une liste séparée par des virgules : Mise à jour critique, mise à jour de sécurité, mise à jour de la définition.

Paramètre	Tapez	Description
SeverityLevels	String (chaîne)	(Facultatif) Spécifiez un ou plusieurs niveaux de sécurité MSRC associés à une mise à jour. Vous pouvez filtrer les niveaux de sécurité en utilisant des valeurs séparées par une virgule. Options: Critique, Important, Faible, Modéré ou Non spécifié. Parmi les formats valides figurent une seule entrée, par exemple : Critique. Vous pouvez également spécifier une liste séparée par des virgules : Critique, Important, Faible.

Étapes d'Automation

Le document `AWS-UpdateWindowsAmi` inclut les étapes d'Automation suivantes par défaut.

Étape 1 : launchInstance(action aws:runInstances)

Cette étape lance une instance avec un rôle de profil d'instance IAM à partir du `SourceAmiID` spécifié.

Étape 2 : runPreUpdateScript(aws:runCommand action)

Cette étape vous permet de spécifier un script comme chaîne exécutée avant l'installation des mises à jour.

Étape 3 : updateEC2Config(aws:runCommand action)

Cette étape utilise le document public AWS-InstallPowerShellModule pour télécharger un AWS module PowerShell public. Systems Manager vérifie l'intégrité du module en utilisant un hachage SHA-256. vérifie Systems Manager ensuite le système d'exploitation pour déterminer s'il faut mettre à jour EC2Config ou EC2Launch . EC2Config s'exécute sur Windows Server 2008 R2 via Windows Server 2012 R2. EC2Launchs'exécute sur Windows Server 2016.

Étape 4 : updateSSMAgent(aws:runCommand action)

Cette étape met à jour Agent SSM en utilisant le document public AWS-UpdateSSMAgent.

Étape 5 : updateAWSPVDriver(aws:runCommand action)

Cette étape met à jour les pilotes AWS PV en utilisant le document public AWS-ConfigureAWSPackage.

Étape 6 : updateAwsEnaNetworkDriver(aws:runCommand action)

Cette étape met à jour les pilotes AWS ENA Network en utilisant le document public AWS-ConfigureAWSPackage.

Étape 7 : installWindowsUpdates(aws:runCommand action)

Cette étape installe les mises à jour Windows en utilisant le document public AWS-InstallWindowsUpdates. Par défaut, Systems Manager recherche et installe l'ensemble des mises à jour manquantes. Vous pouvez modifier le comportement par défaut en spécifiant l'un des paramètres suivants : `IncludeKbs`, `ExcludeKbs` , ou `Categories` .`SeverityLevels`

Étape 8 : runPostUpdateScript(aws:runCommand action)

Cette étape vous permet de spécifier un script comme chaîne qui est exécutée après l'installation des mises à jour.

Étape 9 : runSysprepGeneralize(aws:runCommand action)

Cette étape utilise le document public AWS-InstallPowerShellModule pour télécharger un AWS module PowerShell public. Systems Manager vérifie l'intégrité du module en utilisant un hachage SHA-256. exécute Systems Manager ensuite sysprep à l'aide de méthodes prises en charge par (Windows Server 2016) ou AWS (Windows Server 2008 R2 via 2012 R2).EC2LaunchEC2Config

Étape 10 : stopInstance(action aws:changeInstanceState)

Cette étape arrête toutes les instances mises à jour.

Étape 11 : createImage(action aws:createImage)

Cette étape crée une nouvelle AMI avec un nom descriptif qui la relie à l'ID source et l'heure de création. Exemple : « AMI Generated by EC2 Automation on {{global:DATE_TIME}} from {{SourceAmiId}} » où DATE_TIME et SourceID représentent des variables Automation.

Étape 12 : TerminateInstance(action aws:changeInstanceState)

Cette étape nettoie l'exécution en mettant hors service l'instance en cours d'exécution.

Output

Cette section vous permet de désigner les sorties de différentes étapes ou valeurs de n'importe quel paramètre comme sortie d'Automation. Par défaut, la sortie est l'ID des AMI Windows mises à jour créées par l'exécution.

Note

Par défaut, lorsqu'Automation exécute le document AWS-`AWS-UpdateWindowsAmi` et crée une instance temporaire, le système utilise le VPC par défaut (172.30.0.0/16). Si vous avez supprimé le VPC par défaut, vous recevrez l'erreur suivante :
VPC non défini 400
Pour régler ce problème, vous devez réaliser une copie du document `AWS-UpdateWindowsAmi` et spécifier un ID de sous-réseau. Pour plus d'informations, consultez la section VPC non défini 400 (p. 677).

Pour créer une AMI Windows corrigée à l'aide de l'automatisation

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Exécutez la commande suivante pour exécuter le document `AWS-UpdateWindowsAmi`. Dans la section Paramètres, spécifiez un ID de source d'AMI, un rôle de profil d'instance EC2 et votre rôle de service Automation. L'exemple de commande ci-dessous utilise une AMI Amazon EC2 récente afin de réduire le nombre de correctifs à appliquer. Si vous exécutez cette commande plusieurs fois, vous devez spécifier une valeur unique pour `targetAMIname` . Les noms d'AMI doivent être uniques.

```
aws ssm start-automation-execution --document-name="AWS-UpdateWindowsAmi" --parameters
 SourceAmiId='ami-0246f4914689c475f',IamInstanceProfileName='ManagedInstanceProfile',AutomationAssu
{{global:ACCOUNT_ID}}:role/AutomationServiceRole'
```

   La commande renvoie un ID d'exécution. Copiez cet ID dans le Presse-papiers. Vous allez utiliser cet ID pour afficher l'état du flux de travail.

```
{
    "AutomationExecutionId": "ID"
}
```

3. Pour afficher l'exécution de flux de travail à l'aide de l'AWS CLI, exécutez la commande suivante :

```
aws ssm describe-automation-executions
```

4. Pour afficher les détails de la progression de l'exécution, exécutez la commande suivante.

```
aws ssm get-automation-execution --automation-execution-id ID
```

Note

En fonction du nombre de correctifs appliqués, le processus de mise à jour corrective Windows exécuté dans cet exemple de flux de travail peut prendre 30 minutes ou plus pour se terminer.

Procédure : Simplification de l'application de correctifs à l'AMI avec Automation, AWS Lambda et Parameter Store

L'exemple suivant complète l'exemple de mise à jour d'une AMI Windows, qui est décrit dans Procédure : Appliquer un correctif à une AMI Windows Server (p. 640). Cet exemple utilise le modèle dans lequel une organisation gère et applique périodiquement ses propres correctifs, propriétaires AMIs plutôt que de s'appuyer sur Amazon EC2 AMIs .

La procédure suivante montre comment appliquer automatiquement des correctifs de système d'exploitation à une AMI Windows qui est déjà considérée comme la plus à jour ou à l'AMI la plus récente. Dans l'exemple, la valeur par défaut du paramètre `SourceAmiId` est définie par un Systems Manager Parameter Store paramètre appelé `latestAmi` . La valeur de `latestAmi` est mise à jour par une AWS Lambda fonction appelée à la fin du flux de travail Automation. Suite à ce processus Automation, le temps et les efforts consacrés à appliquer les correctifs AMIs sont minimisés, car ils sont toujours appliqués à l'AMI la plus récente.

Avant de commencer

Configurez des rôles et, éventuellement, des EventBridge pour Automation. Pour plus d'informations, consultez la section Configuration d'Automation (p. 308).

Table des matières

- Tâche 1 : Créer un paramètre dansSystems ManagerParameter Store (p. 645)
- Tâche 2 : Création d'un IAM rôle pourAWS Lambda (p. 645)
- Tâche 3 : Création d'une AWS Lambda fonction (p. 647)
- Tâche 4 : Créer un document Automation et corriger l'AMI (p. 648)

Tâche 1 : Créer un paramètre dansSystems ManagerParameter Store

Créez un paramètre dans Parameter Store qui utilise les informations suivantes :

- Nom : `latestAmi`.
- Valeur : ID d'une AMI Windows. Par exemple : `ami-188d6e0e`.

Pour de plus amples informations sur la création d'un paramètre de chaîne Parameter Store, veuillez consulter Création de paramètres Systems Manager (p. 243).

Tâche 2 : Création d'un IAM rôle pourAWS Lambda

Utilisez la procédure suivante pour créer un rôle de service IAM pour AWS Lambda. Ce rôle inclut les stratégies gérées `AmazonSSMFullAccess` et `AWSLambdaExecute`. Ces stratégies autorisent Lambda à mettre à jour la valeur du paramètre `latestAmi` à l'aide d'une fonction Lambda et Systems Manager.

Pour créer un rôle de service IAM pour Lambda

1. Connectez-vous à la AWS Management Console et ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation, choisissez Stratégies, puis Créer une stratégie.
3. Choisissez l'onglet JSON.
4. Remplacez le contenu par défaut par la commande suivante : Veillez à remplacer *us-west-2* et *123456789012* avec la région et le compte que vous souhaitez utiliser. Remplacez *updateAmiFunction* par le nom de votre Lambda fonction.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "logs:CreateLogGroup",
            "Resource": "arn:aws:logs:us-west-2:123456789012:*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogStream",
                "logs:PutLogEvents"
            ],
            "Resource": [
                "arn:aws:logs:us-west-2:123456789012:log-group:/aws/
lambda/updateAmiFunction:*"
            ]
        }
    ]
}
```

5. Choisissez Examiner une stratégie.
6. Sur la page Examiner une stratégie, dans le champ Nom, saisissez un nom pour la stratégie en ligne, tel que **amiLambda**.
7. Choisissez Create policy (Créer une stratégie).
8. Répétez les étapes 2 et 3.
9. Remplacez le contenu par défaut par la commande suivante : Veillez à remplacer *us-west-2* et *123456789012* avec la région et le compte que vous souhaitez utiliser.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ssm:PutParameter",
            "Resource": "arn:aws:ssm:us-west-2:123456789012:parameter/latestAmi"
        },
        {
            "Effect": "Allow",
            "Action": "ssm:DescribeParameters",
            "Resource": "*"
        }
    ]
}
```

10. Choisissez Examiner une stratégie.
11. Sur la page Examiner une stratégie, dans le champ Nom, saisissez un nom pour la stratégie en ligne, tel que **amiParameter**.

12. Choisissez Create policy (Créer une stratégie).
13. Dans le volet de navigation, choisissez Rôles, puis Créer un rôle.
14. Pour Choose the service that will use this role (Sélectionner immédiatement le service qui utilisera ce rôle), choisissez Lambda, puis Next (Suivant) : Autorisations.
15. Dans la page Attacher des stratégies d'autorisations utilisez le champ Recherche pour localiser les deux stratégies que vous avez créées précédemment.
16. Cochez la case en regard des stratégies, puis choisissez Suivant : Balises.
17. (Facultatif) Ajoutez une ou plusieurs paires clé-valeur de balise pour organiser, suivre ou contrôler l'accès pour ce rôle, puis choisissez Suivant : Passez en revue.
18. Pour Nom du rôle, entrez un nom pour votre nouveau rôle, par exemple **`lambda-ssm-role`** ou autre, en fonction de vos préférences.

    Note

    Différentes entités peuvent référencer le rôle et il n'est donc pas possible de modifier son nom après sa création.
19. Sélectionnez Créer un rôle.

Tâche 3 : Création d'une AWS Lambda fonction

Utilisez la procédure suivante pour créer une fonction Lambda qui met à jour automatiquement la valeur du paramètre `latestAmi`.

Pour créer une fonction Lambda

1. Connectez-vous à la AWS Management Console et ouvrez la console AWS Lambda à l'adresse https://console.aws.amazon.com/lambda/.
2. Sélectionnez Create function.
3. Sur la page Créer une fonction, choisissez Créer à partir de zéro.
4. Pour Nom de la fonction, entrez **`Automation-UpdateSsmParam`**.
5. Dans la liste Runtime, choisissez Python 3.8.
6. Dans la section Autorisations, développez Choisir ou créer un rôle d'exécution.
7. Choisissez Utiliser un rôle existant, puis choisissez le rôle de service Lambda que vous avez créé lors de la tâche 2.
8. Sélectionnez Create function.
9. Dans la section Code de fonction, supprimez le code prérempli dans le champ lambda_function, puis collez l'exemple de code suivant.

```
from __future__ import print_function

import json
import boto3

print('Loading function')


#Updates an SSM parameter
#Expects parameterName, parameterValue
def lambda_handler(event, context):
    print("Received event: " + json.dumps(event, indent=2))

    # get SSM client
    client = boto3.client('ssm')

    #confirm  parameter exists before updating it
```

```
    response = client.describe_parameters(
       Filters=[
          {
           'Key': 'Name',
           'Values': [ event['parameterName'] ]
          },
        ]
    )

    if not response['Parameters']:
        print('No such parameter')
        return 'SSM parameter not found.'

    #if parameter has a Description field, update it PLUS the Value
    if 'Description' in response['Parameters'][0]:
        description = response['Parameters'][0]['Description']

        response = client.put_parameter(
          Name=event['parameterName'],
          Value=event['parameterValue'],
          Description=description,
          Type='String',
          Overwrite=True
        )

    #otherwise just update Value
    else:
        response = client.put_parameter(
          Name=event['parameterName'],
          Value=event['parameterValue'],
          Type='String',
          Overwrite=True
        )

    reponseString = 'Updated parameter %s with value %s.' % (event['parameterName'],
 event['parameterValue'])

    return reponseString
```

10. Choisissez Save.
11. Pour tester la fonction Lambda, dans le menu Événements de test, choisissez Configurer des événements de test.
12. Pour Event name (Nom d'événement), saisissez un nom pour l'événement de test, tel que **`MyTestEvent`**.
13. Remplacez le texte existant par le JSON suivant, en remplaçant *`your-ami-id`* avec l'ID de la nouvelle AMI à définir comme valeur de `latestAmi` paramètre.

```
{
   "parameterName":"latestAmi",
   "parameterValue":"your-ami-id"
}
```

14. Sélectionnez Create.
15. Sélectionnez Test pour tester la fonction. La sortie doit indiquer que le paramètre a été mis à jour correctement et inclure des détails sur la mise à jour. Par exemple, « Paramètre mis à jour latestAmi avec la valeur ami-123456 ».

Tâche 4 : Créer un document Automation et corriger l'AMI

Utilisez la procédure suivante pour créer et exécuter un document Automation qui applique des correctifs à l'AMI que vous avez spécifiée pour le latestAmi paramètre. Une fois le flux de travail Automation terminé,

la valeur de latestAmi est mise à jour avec l'ID de l'AMI nouvellement corrigée. Les exécutions ultérieures utilisent l'AMI créée par l'exécution précédente.

Pour créer un document d'Automation et appliquer un correctif à une AMI

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Documents dans le panneau de navigation.
3. Choisissez Create automation (Créer une automatisation).
4. Dans le champ Name, saisissez **`UpdateMyLatestWindowsAmi`**.
5. Choisissez l'onglet Éditeur, puis Modifier.
6. Remplacez le contenu par défaut dans le champ Éditeur de document avec l'exemple de document JSON suivant.

   > **Note**
   >
   > Vous devez modifier les valeurs de *`assumeRole`* et *`IamInstanceProfileName`* dans cet exemple, avec l'ARN du rôle de service et le rôle de profil d'instance que vous avez créés lors Configuration d'Automation (p. 308) de la création de .

```
{
   "description":"Systems Manager Automation Demo - Patch AMI and Update SSM Param",
   "schemaVersion":"0.3",
   "assumeRole":"the role ARN you created",
   "parameters":{
      "sourceAMIid":{
         "type":"String",
         "description":"AMI to patch",
         "default":"{{ssm:latestAmi}}"
      },
      "targetAMIname":{
         "type":"String",
         "description":"Name of new AMI",
         "default":"patchedAMI-{{global:DATE_TIME}}"
      }
   },
   "mainSteps":[
      {
         "name":"startInstances",
         "action":"aws:runInstances",
         "timeoutSeconds":1200,
         "maxAttempts":1,
         "onFailure":"Abort",
         "inputs":{
            "ImageId":"{{ sourceAMIid }}",
            "InstanceType":"m3.large",
            "MinInstanceCount":1,
            "MaxInstanceCount":1,
            "IamInstanceProfileName":"the name of the IAM role you created"
         }
      },
      {
         "name":"installMissingWindowsUpdates",
         "action":"aws:runCommand",
         "maxAttempts":1,
         "onFailure":"Continue",
```

```
         "inputs":{
            "DocumentName":"AWS-InstallWindowsUpdates",
            "InstanceIds":[
               "{{ startInstances.InstanceIds }}"
            ],
            "Parameters":{
               "SeverityLevels":"Important"
            }
         }
      },
      {
         "name":"stopInstance",
         "action":"aws:changeInstanceState",
         "maxAttempts":1,
         "onFailure":"Continue",
         "inputs":{
            "InstanceIds":[
               "{{ startInstances.InstanceIds }}"
            ],
            "DesiredState":"stopped"
         }
      },
      {
         "name":"createImage",
         "action":"aws:createImage",
         "maxAttempts":1,
         "onFailure":"Continue",
         "inputs":{
            "InstanceId":"{{ startInstances.InstanceIds }}",
            "ImageName":"{{ targetAMIname }}",
            "NoReboot":true,
            "ImageDescription":"AMI created by EC2 Automation"
         }
      },
      {
         "name":"terminateInstance",
         "action":"aws:changeInstanceState",
         "maxAttempts":1,
         "onFailure":"Continue",
         "inputs":{
            "InstanceIds":[
               "{{ startInstances.InstanceIds }}"
            ],
            "DesiredState":"terminated"
         }
      },
      {
         "name":"updateSsmParam",
         "action":"aws:invokeLambdaFunction",
         "timeoutSeconds":1200,
         "maxAttempts":1,
         "onFailure":"Abort",
         "inputs":{
            "FunctionName":"Automation-UpdateSsmParam",
            "Payload":"{\"parameterName\":\"latestAmi\", \"parameterValue\":
\"{{createImage.ImageId}}\"}"
         }
      }
   ],
   "outputs":[
      "createImage.ImageId"
   ]
}
```

7. Choisissez Create automation (Créer une automatisation) pour enregistrer le document.

8. Dans le panneau de navigation, choisissez Automation (Automatisation), puis Execute automation (Exécuter l'automatisation).
9. Dans la page Choose document (Choisir un document), choisissez l'onglet Appartenant à moi, puis sélectionnez le bouton dans la UpdateMyLatestWindowsAmi carte.
10. Dans la section Document details (Détails du document), vérifiez que l'option Document version (Version du document) est définie sur 1 (Default) (1 (Par défaut)).
11. Choisissez Nex t(Suivant).
12. Choisissez Exécution simple.
13. Sélectionnez Exécuter.
14. Une fois l'exécution terminée, choisissez Parameter Store dans le panneau de navigation et vérifiez que la nouvelle valeur de `latestAmi` correspond à celle renvoyée par le flux de travail d'automatisation. Vous pouvez également vérifier que l'ID de la nouvelle AMI correspond à la sortie d'Automation dans la AMIs section de la Amazon EC2 console.

Procédure : Appliquer des correctifs à une AMI et mettre à jour un groupe Auto Scaling

L'exemple suivant repose sur l'Procédure : Simplification de l'application de correctifs à l'AMI avec Automation, AWS Lambda et Parameter Store (p. 645)exemple en ajoutant une étape qui met à jour un Auto Scaling groupe avec l'AMI nouvellement corrigée. Cette approche permet de s'assurer que des nouvelles images sont automatiquement mises à disposition dans différents environnements de calcul qui utilisent des groupes Auto Scaling.

L'étape finale du flux de travail Automation dans cet exemple utilise une AWS Lambda fonction pour copier une configuration de lancement existante et définir l'ID d'AMI sur l'AMI nouvellement corrigée. Le groupe Auto Scaling est alors mis à jour avec la nouvelle configuration de lancement. Dans ce type de scénario Auto Scaling, les utilisateurs peuvent mettre hors fonction des instances existantes dans le groupe Auto Scaling pour forcer une nouvelle instance utilisant la nouvelle image à se lancer. Ou les utilisateurs peuvent attendre et permettre aux événements de mise à l'échelle vers le haut ou vers le bas de lancer naturellement des instances plus récentes.

Avant de commencer

Exécutez les tâches suivantes avant de commencer cet exemple.

- Configurez des rôles IAM pour Automation. Systems Manager nécessite un rôle de profil d'instance et un ARN de rôle de service pour traiter les flux de travail d'automatisation. Pour plus d'informations, consultez la section Configuration d'Automation (p. 308).
- Si vous n'êtes pas familiarisé avec Lambda, nous vous recommandons de créer une fonction Lambda simple à l'aide de la rubrique Création d'une fonction Lambda simple du AWS Lambda Developer Guide. Cette rubrique vous aidera à comprendre en détails certaines des étapes nécessaires pour créer une fonction Lambda.

Tâche 1 : Création d'un IAM rôle pourAWS Lambda

Utilisez la procédure suivante pour créer un rôle de service IAM pour AWS Lambda. Ce rôle inclut les stratégies gérées AutoScalingFullAccess et AWSLambdaExecute. Ces stratégies autorisent Lambda à créer un nouveau groupe Auto Scaling avec la dernière AMI corrigée à l'aide d'une fonction Lambda.

Pour créer un rôle de service IAM pour Lambda

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le volet de navigation, choisissez Rôles, puis Créer un rôle.
3. Sur la page Select type of trusted entity (Sélectionner le type d'entité de confiance), choisissez AWS Service (Service AWS).

4. Dans la section Choose a use case (Choisir un cas d'utilisation), choisissez Lambda , puis Next (Suivant) : Autorisations.
5. Sur la page Attach permissions policies (Attacher des stratégies d'autorisationsAWSLambdaExecute), recherchez , puis choisissez l'option située en regard. Recherchez AutoScalingFullAccess , puis choisissez l'option en regard de celui-ci.
6. Choisissez Next (Suivant) : Balises.
7. (Facultatif) Ajoutez une ou plusieurs paires clé-valeur de balise pour organiser, suivre ou contrôler l'accès pour ce rôle, puis choisissez Suivant : Passez en revue.
8. Sur la page Vérifier, vérifiez que AWSLambdaExecute et AutoScalingFullAccess sont répertoriés sous Stratégies.

9. Entrez un nom dans la zone Role name (Nom de rôle), puis saisissez une description.
10. Sélectionnez Créer un rôle. Le système vous renvoie à la page Rôles.

Tâche 2 : Création d'une AWS Lambda fonction

Utilisez la procédure suivante pour créer une fonction Lambda qui met à jour automatiquement un groupe Auto Scaling existant avec la dernière AMI corrigée.

Pour créer une fonction Lambda

1. Connectez-vous à la AWS Management Console et ouvrez la console AWS Lambda à l'adresse https://console.aws.amazon.com/lambda/.
2. Sélectionnez Create function.
3. Vérifiez que l'option Créer à partir de zéro est sélectionnée.
4. Dans le champ Nom de la fonction, entrez **`Automation-UpdateAsg`** .
5. Dans la liste Runtime, choisissez Python 2.7.
6. Développez Choisissez ou créez un rôle d'exécution puis, dans la liste Execution role (Rôle d'exécution), vérifiez que Use an existing role (Utiliser un rôle existant) est sélectionné.
7. Dans la liste Rôle existant, choisissez le rôle que vous avez créé précédemment.
8. Sélectionnez Create function. Le système affiche un code et une page de configuration pour Automation-UpdateAsg.
9. N'apportez aucune modification dans la section Designer.
10. Dans la section Function code (Code de fonction), supprimez le code prérempli dans le champ lambda_function, puis collez l'exemple de code suivant.

```
from __future__ import print_function

import json
```

```
import datetime
import time
import boto3

print('Loading function')


def lambda_handler(event, context):
    print("Received event: " + json.dumps(event, indent=2))

    # get autoscaling client
    client = boto3.client('autoscaling')

    # get object for the ASG we're going to update, filter by name of target ASG
    response =
 client.describe_auto_scaling_groups(AutoScalingGroupNames=[event['targetASG']])

    if not response['AutoScalingGroups']:
        return 'No such ASG'

    # get name of InstanceID in current ASG that we'll use to model new Launch
 Configuration after
    sourceInstanceId = response.get('AutoScalingGroups')[0]['Instances'][0]
['InstanceId']

    # create LC using instance from target ASG as a template, only diff is the name of
 the new LC and new AMI
    timeStamp = time.time()
    timeStampString = datetime.datetime.fromtimestamp(timeStamp).strftime('%Y-%m-%d
 %H-%M-%S')
    newLaunchConfigName = 'LC '+ event['newAmiID'] + ' ' + timeStampString
    client.create_launch_configuration(
        InstanceId = sourceInstanceId,
        LaunchConfigurationName=newLaunchConfigName,
        ImageId= event['newAmiID'] )

    # update ASG to use new LC
    response = client.update_auto_scaling_group(AutoScalingGroupName =
 event['targetASG'],LaunchConfigurationName = newLaunchConfigName)

    return 'Updated ASG `%s` with new launch configuration `%s` which includes AMI `
%s`.' % (event['targetASG'], newLaunchConfigName, event['newAmiID'])
```

11. Spécifiez les options de configuration restantes sur cette page.
12. Choisissez Save.
13. Sélectionnez Test.
14. Dans la page Configure test event (Configurer un événement de test), vérifiez que Create new test event (Créer un nouvel événement de test) est sélectionné.
15. Dans la liste Event template (Modèle d'événement), vérifiez que Hello World est sélectionné.
16. Dans le champ Event name (Nom de l'événement), entrez un nom.
17. Remplacez l'exemple existant par le code JSON suivant. Entrez un ID d'AMI et un groupe Auto Scaling.

```
{
   "newAmiID":"valid AMI ID",
   "targetASG":"name of your Auto Scaling group"
}
```

18. Sélectionnez Create.
19. Sélectionnez Test. La sortie indique que le groupe Auto Scaling a été mis à jour avec la nouvelle configuration de lancement.

Tâche 3 : Créer un document Automation, corriger l'AMI et mettre à jour le Auto Scaling groupe

Utilisez la procédure suivante pour créer et exécuter un document Automation qui corrige l'AMI que vous avez spécifiée pour le latestAmi paramètre. Le flux de travail d'automatisation met alors à jour le groupe Auto Scaling pour utiliser la dernière AMI corrigée.

Pour créer et exécuter un document d'automatisation

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Documents dans le panneau de navigation.
3. Choisissez Create automation (Créer une automatisation).
4. Dans le champ Nom, saisissez **`PatchAmiandUpdateAsg`**.
5. Cliquez sur l'onglet Editor (Éditeur) puis sur le bouton Edit (Modifier).
6. Choisissez OK lorsque vous y êtes invité et supprimez le contenu de l'espace réservé dans le champ Document editor (Éditeur de documents).
7. Dans le champ Document editor (Éditeur de documents) collez l'exemple de contenu de document JSON suivant.

   > **Note**
   >
   > Vous devez modifier les valeurs de *`assumeRole`* et *`IamInstanceProfileName`* dans cet exemple, avec l'ARN du rôle de service et le rôle de profil d'instance que vous avez créés lors Configuration d'Automation (p. 308) de la création de .

```
{
   "description":"Systems Manager Automation Demo - Patch AMI and Update ASG",
   "schemaVersion":"0.3",
   "assumeRole":"the service role ARN you created",
   "parameters":{
      "sourceAMIid":{
         "type":"String",
         "description":"AMI to patch"
      },
      "subnetId":{
        "type":"String",
        "description":"The SubnetId where the instance is launched from the
 sourceAMIid."
      },
      "targetAMIname":{
         "type":"String",
         "description":"Name of new AMI",
         "default":"patchedAMI-{{global:DATE_TIME}}"
      },
     "targetASG":{
         "type":"String",
         "description":"Auto Scaling group to Update"
      }
   },
   "mainSteps":[
      {
         "name":"startInstances",
         "action":"aws:runInstances",
         "timeoutSeconds":1200,
         "maxAttempts":1,
```

```
            "onFailure":"Abort",
            "inputs":{
               "ImageId":"{{ sourceAMIid }}",
               "InstanceType":"m3.large",
               "MinInstanceCount":1,
               "MaxInstanceCount":1,
               "IamInstanceProfileName":"the name of the instance IAM role you created",
               "SubnetId":"{{ subnetId }}"
            }
         },
         {
            "name":"installMissingWindowsUpdates",
            "action":"aws:runCommand",
            "maxAttempts":1,
            "onFailure":"Continue",
            "inputs":{
               "DocumentName":"AWS-InstallMissingWindowsUpdates",
               "InstanceIds":[
                  "{{ startInstances.InstanceIds }}"
               ],
               "Parameters":{
                  "UpdateLevel":"Important"
               }
            }
         },
         {
            "name":"stopInstance",
            "action":"aws:changeInstanceState",
            "maxAttempts":1,
            "onFailure":"Continue",
            "inputs":{
               "InstanceIds":[
                  "{{ startInstances.InstanceIds }}"
               ],
               "DesiredState":"stopped"
            }
         },
         {
            "name":"createImage",
            "action":"aws:createImage",
            "maxAttempts":1,
            "onFailure":"Continue",
            "inputs":{
               "InstanceId":"{{ startInstances.InstanceIds }}",
               "ImageName":"{{ targetAMIname }}",
               "NoReboot":true,
               "ImageDescription":"AMI created by EC2 Automation"
            }
         },
         {
            "name":"terminateInstance",
            "action":"aws:changeInstanceState",
            "maxAttempts":1,
            "onFailure":"Continue",
            "inputs":{
               "InstanceIds":[
                  "{{ startInstances.InstanceIds }}"
               ],
               "DesiredState":"terminated"
            }
         },
         {
            "name":"updateASG",
            "action":"aws:invokeLambdaFunction",
            "timeoutSeconds":1200,
            "maxAttempts":1,
```

```
            "onFailure":"Abort",
            "inputs": {
               "FunctionName": "Automation-UpdateAsg",
               "Payload": "{\"targetASG\":\"{{targetASG}}\", \"newAmiID\":
\"{{createImage.ImageId}}\"}"
            }
         }
      ],
      "outputs":[
         "createImage.ImageId"
      ]
}
```

8. Choisissez Create document (Créer un document) pour enregistrer le document.
9. Choisissez Automations (Automatisations), puis Execute automation (Exécuter l'automatisation).
10. Dans la liste des documents Automation, choisissez PatchAmiandUpdateAsg .
11. Dans la section Détails du document, vérifiez que l'option Version du document est définie sur 1.
12. Dans la section Execution mode (Mode d'exécution), choisissez Execute the entire automation at once (Exécuter l'automatisation complète en une seule fois).
13. Laissez l'option Targets and Rate Control (Cibles et contrôle du débit) désactivée.
14. Spécifiez un ID d'AMI Windows pour **`sourceAMIid`**, le nom de votre groupe Auto Scaling pour **`targetASG`** et une valeur pour le paramètre d'entrée **`subnetId`**.
15. Choisissez Execute automation (Exécuter l'automatisation).
16. Une fois l'exécution terminée dans la console Amazon EC2, choisissez Auto Scaling, puis Launch Configurations (Lancer les configurations). Vérifiez que vous voyez la nouvelle configuration du lancement et que celle-ci utilise l'ID de la nouvelle AMI.
17. Choisissez Auto Scaling, puis Auto Scaling Groups. Vérifiez que le groupe Auto Scaling utilise la nouvelle configuration de lancement.
18. Mettez hors fonction une ou plusieurs instances dans votre groupe Auto Scaling. Des instances de remplacement seront lancées avec l'ID de votre nouvelle AMI.

    Note

    Vous pouvez encore plus automatiser le déploiement de la nouvelle AMI en modifiant la fonction Lambda pour mettre hors service normalement les instances. Vous pouvez également appeler votre propre fonction Lambda et exploiter la capacité d'AWS CloudFormation à mettre à jour les groupes Auto Scaling. Pour plus d'informations, consultez UpdatePolicy Attribut.

Utilisation des instances Automation en libre-service du support AWS

Cette section décrit comment exécuter des instances d'Automation créés par l'équipe AWS Support pour vous aider à résoudre les problèmes courants avec vos ressources AWS.

Rubriques

- Procédure : Exécuter l'EC2Rescueoutil sur des instances inaccessibles (p. 656)
- Procédure : Réinitialiser les mots de passe et les clés SSH sur les instances EC2 (p. 661)

Procédure : Exécuter l'EC2Rescueoutil sur des instances inaccessibles

EC2Rescuepeut vous aider à diagnostiquer et à résoudre les problèmes qui peuvent survenir sur les instances EC2 pour Linux et Windows Server . Vous pouvez exécuter l'outil manuellement, comme décrit dans Utilisation de EC2Rescue pour Linux Server et Utilisation EC2Rescue de pour Windows Server. Vous pouvez également exécuter l'outil automatiquement à l'aide Systems Manager d'Automation et du

document AWSSupport -ExecuteEC2Rescue. Le document AWSSupport -ExecuteEC2Rescue est conçu pour exécuter une combinaison d'Systems Manageractions , d'AWS CloudFormationactions et de Lambda fonctions qui automatisent les étapes normalement requises pour utiliser EC2Rescue .

Vous pouvez utiliser le document AWSSupport -ExecuteEC2Rescue pour dépanner et potentiellement corriger différents types de problèmes liés aux systèmes d'exploitation. Pour obtenir une liste complète, veuillez consulter les rubriques suivantes :

Windows: avec la ligne de commande, consultez Action de résolution dans l'utilisation EC2Rescue d' pour Windows Server.

Linux : Certains modules EC2Rescue pour Linux détectent et tentent de corriger des problèmes. Pour plus d'informations, consultez la documentation aws-ec2rescue-linux pour chaque module sur GitHub .

Fonctionnement

Le dépannage d'une instance avec Automation et le document AWSSupport -ExecuteEC2Rescue fonctionne comme suit :

- Vous spécifiez l'ID de l'instance inaccessible et exécutez le flux de travail d'automatisation.
- Le système crée un VPC temporaire, puis exécute une série de fonctions Lambda pour configurer le VPC.
- Le système identifie un sous-réseau pour votre VPC temporaire dans la même zone de disponibilité que votre instance d'origine.
- Le système lance une instance d'assistant SSM temporaire.
- Le système arrête l'instance originale et crée une sauvegarde. Ensuite, il rattache le volume racine original à l'instance d'assistant.
- Le système utilise Fonctionnalité Exécuter la commande pour s'exécuter EC2Rescue sur l'instance d'assistant. EC2Rescueidentifie et tente de corriger les problèmes sur le volume racine original attaché. Une fois terminé, EC2Rescue rattache le volume racine à l'instance d'origine.
- Le système redémarre votre instance originale et met fin à l'instance temporaire. Le système met également fin au VPC temporaire et aux fonctions Lambda créés au début de l'automatisation.

Avant de commencer

Avant d'exécuter l'automatisation suivante, veillez à exécuter les actions suivantes :

- Copiez l'ID de l'instance de l'instance inaccessible. Vous spécifierez cet ID au cours de la procédure.
- Vous pouvez également recueillir l'ID d'un sous-réseau dans la même zone de disponibilité que votre instance inaccessible. LEC2Rescue'instance sera créée dans ce sous-réseau. Si vous ne spécifiez pas de sous-réseau, Automation crée un nouveau VPC temporaire dans votre compte AWS. Vérifiez que votre compte AWS possède au moins un VPC disponible. Par défaut, vous pouvez en créer cinq VPCs dans une région. Si vous en avez déjà créé cinq VPCs dans la région, l'automatisation échoue sans apporter de modifications à votre instance. Pour plus d'informations sur les Amazon VPC quotas, consultez VPC et sous-réseaux dans le Amazon VPC Guide de l'utilisateur .
- Vous pouvez éventuellement créer et spécifier un rôle AWS Identity and Access Management (IAM) pour l'automatisation. Si vous ne spécifiez pas ce rôle, Automation s'exécute dans le contexte de l'utilisateur qui a exécuté l'automatisation. Pour de plus amples informations sur la création de rôles pour l'automatisation, veuillez consulter Exécution d'une automatisation à l'aide d'un rôle IAM de service (p. 368).

Attribution d'autorisations AWSSupport -EC2Rescue pour effectuer des actions sur vos instances

EC2Rescuedoit être autorisé à effectuer une série d'actions sur vos instances au cours de l'exécution d'Automation. Ces actions appellent les services AWS Lambda, IAM et Amazon EC2 pour essayer en

toute sécurité de corriger les problèmes liés à vos instances. Si vous avez des autorisations niveau administrateur dans votre compte AWS et/ou VPC, vous pouvez peut-être exécuter l'instance d'Automation sans configurer les autorisations, comme décrit dans cette section. Si vous ne possédez pas les autorisations niveau administrateur, donc vous, ou un administrateur, devez configurer les autorisations en utilisant l'une des options suivantes.

- Attribution des autorisations en utilisant les stratégies IAM (p. 658)
- Attribution des autorisations en utilisant un modèle AWS CloudFormation (p. 659)

Attribution des autorisations en utilisant les stratégies IAM

Vous pouvez attacher la stratégie IAM suivante à votre compte utilisateur, groupe ou rôle IAM en tant que stratégie en ligne, ou vous pouvez créer une nouvelle stratégie IAM gérée et la relier à votre compte utilisateur, groupe ou rôle. Pour plus d'informations au sujet de l'ajout d'une stratégie en ligne à votre compte utilisateur, groupe ou rôle, consultez la page Utilisation de stratégies en ligne. Pour plus d'informations sur la création d'une nouvelle stratégie gérée, consultez la page Utilisation de stratégies gérées.

> **Note**
>
> Si vous créez une nouvelle stratégie IAM gérée, vous devez également lui attacher la stratégie AmazonSSMAutomationRole gérée afin que vos instances puissent communiquer avec l'Systems ManagerAPI.

IAM Stratégie pour AWSSupport -EC2Rescue

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Action": [
            "lambda:InvokeFunction",
            "lambda:DeleteFunction",
            "lambda:GetFunction"
         ],
         "Resource": "arn:aws:lambda:*:aws-account-id:function:AWSSupport-EC2Rescue-*",
         "Effect": "Allow"
      },
      {
         "Action": [
            "s3:GetObject",
            "s3:GetObjectVersion"
         ],
         "Resource": [
            "arn:aws:s3:::awssupport-ssm.*/*.template",
            "arn:aws:s3:::awssupport-ssm.*/*.zip"
         ],
         "Effect": "Allow"
      },
      {
         "Action": [
            "iam:CreateRole",
            "iam:CreateInstanceProfile",
            "iam:GetRole",
            "iam:GetInstanceProfile",
            "iam:PutRolePolicy",
            "iam:DetachRolePolicy",
            "iam:AttachRolePolicy",
            "iam:PassRole",
            "iam:AddRoleToInstanceProfile",
            "iam:RemoveRoleFromInstanceProfile",
            "iam:DeleteRole",
```

```
            "iam:DeleteRolePolicy",
            "iam:DeleteInstanceProfile"
         ],
         "Resource": [
            "arn:aws:iam::aws-account-id:role/AWSSupport-EC2Rescue-*",
            "arn:aws:iam::aws-account-id:instance-profile/AWSSupport-EC2Rescue-*"
         ],
         "Effect": "Allow"
      },
      {
         "Action": [
            "lambda:CreateFunction",
            "ec2:CreateVpc",
            "ec2:ModifyVpcAttribute",
            "ec2:DeleteVpc",
            "ec2:CreateInternetGateway",
            "ec2:AttachInternetGateway",
            "ec2:DetachInternetGateway",
            "ec2:DeleteInternetGateway",
            "ec2:CreateSubnet",
            "ec2:DeleteSubnet",
            "ec2:CreateRoute",
            "ec2:DeleteRoute",
            "ec2:CreateRouteTable",
            "ec2:AssociateRouteTable",
            "ec2:DisassociateRouteTable",
            "ec2:DeleteRouteTable",
            "ec2:CreateVpcEndpoint",
            "ec2:DeleteVpcEndpoints",
            "ec2:ModifyVpcEndpoint",
            "ec2:Describe*"
         ],
         "Resource": "*",
         "Effect": "Allow"
      }
   ]
}
```

Attribution des autorisations en utilisant un modèle AWS CloudFormation

AWS CloudFormation automatise le processus de création des rôles et stratégies IAM en utilisant un modèle préconfiguré. Utilisez la procédure suivante pour créer les IAM rôles et stratégies requis pour l'EC2Rescueautomatisation à l'aide d' AWS CloudFormation .

Pour créer les IAM rôles et stratégies requis pourEC2Rescue

1. Téléchargez `AWSSupport-EC2RescueRole.zip` et extrayez le fichier `AWSSupport-EC2RescueRole.json` pour l'enregistrer dans un répertoire de votre machine locale.
2. Si votre compte AWS est une partition spéciale, modifiez le modèle pour remplacer les valeurs de l'ARN par celles de votre partition.

   Par exemple, pour les Régions de Chine, remplacez toutes les occurrences de `arn:aws` par `arn:aws-cn`.
3. Connectez-vous à la AWS Management Console et ouvrez la console AWS CloudFormation à l'adresse https://console.aws.amazon.com/cloudformation.
4. Choisissez Créer une pile, Avec de nouvelles ressources (standard).
5. Sur la page Créer une pile, pour Prérequis - Préparer le modèle, choisissez Le modèle est prêt.
6. Dans Spécifier le modèle, choisissez Charger un modèle de fichier.
7. Sélectionnez Choisir le fichier, puis recherchez et sélectionnez le fichier `AWSSupport-EC2RescueRole.json` dans le répertoire où vous l'avez extrait.

8. Choisissez Nex t(Suivant).
9. Dans la page Spécifier les détails de la pile, pour le champ Nom de la pile, entrez un nom pour identifier cette pile, puis choisissez Suivant.
10. (Facultatif) Dans la zone Balises, appliquez une ou plusieurs paires nom/valeur de clé de balise à la pile.

    Les balises sont des métadonnées facultatives que vous affectez à une ressource. Les balises vous permettent de classer une ressource de différentes façons, par exemple, par objectif, par propriétaire ou par environnement. Par exemple, vous pouvez baliser une pile pour identifier le type de tâches qu'elle exécute, les types de cibles ou d'autres ressources concernées, et l'environnement dans lequel elle est exécutée.
11. Choisissez Suivant.
12. Sur la page Révision, passez en revue les détails de la pile, puis faites défiler vers le bas et choisissez l'option Je reconnais que AWS CloudFormation pourrait créer des ressources IAM.
13. Choisissez Créer une pile.

    AWS CloudFormation affiche l'état CREATE_IN_PROGRESS pendant quelques minutes. L'état passe à CREATE_COMPLETE après la création de la pile. Vous pouvez également choisir l'icône d'actualisation pour vérifier l'état du processus de création.
14. Dans la liste Piles, choisissez l'option à côté de la pile que vous venez de créer, puis sélectionnez l'onglet Sorties.
15. Notez la Valeur. est l'ARN de l' AssumeRole . Vous spécifiez cet ARN lorsque vous exécutez Automation dans la procédure suivante, Exécution d'Automation (p. 660) .

Exécution d'Automation

> **Important**
>
> L'exécution du flux de travail d'automatisation suivant arrête l'instance inaccessible. L'arrêt de l'instance peut entraîner la perte de données sur des volumes de stockage d'instance attachés (le cas échéant). L'arrêt de l'instance peut aussi causer le changement de l'adresse IP publique, si aucune adresse IP Elastic n'est associée.

Pour exécuter l'automatisation AWSSupport -ExecuteEC2Rescue

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Automatisation.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Automation.
3. Choisissez Execute automation (Exécuter l'automatisation).
4. Dans la section Document d'automatisation, choisissez M'appartenant ou appartenant à Amazon dans la liste.
5. Dans la liste des documents, choisissez le bouton de la carte pour AWSSupport -ExecuteEC2Rescue, puis choisissez Next (Suivant).
6. Sur la page Exécuter le document d'automatisation, choisissez Exécution simple.
7. Dans la section Détails du document, vérifiez que l'option Version du document est définie sur la version par défaut la plus importante. Par exemple, $DEFAULT ou 3 (par défaut).
8. Dans la section Paramètres d'entrée, spécifiez les paramètres suivants :

   a. Pour UnreachableInstanceId , spécifiez l'ID de l'instance inaccessible.

b. (Facultatif) Pour EC2RescueInstanceType , spécifiez un type pour l'EC2Rescueinstance. Le type d'instance par défaut est `t2.small`.
c. Pour AutomationAssumeRole , si vous avez créé des rôles pour cette automatisation à l'aide de la AWS CloudFormation procédure décrite précédemment dans cette rubrique, choisissez l'ARN de l' AssumeRole que vous avez créé dans la AWS CloudFormation console .
d. (Facultatif) Pour LogDestination , spécifiez un compartiment S3 si vous souhaitez collecter les journaux au niveau du système d'exploitation tout en dépannant votre instance. Les journaux sont chargés automatiquement dans le compartiment spécifié.
e. Pour SubnetId , spécifiez un sous-réseau dans un VPC existant dans la même zone de disponibilité que l'instance inaccessible. Par défaut, Systems Manager crée un VPC, mais vous pouvez spécifier un sous-réseau dans un VPC existant si vous le souhaitez.

   Note

   Si vous ne voyez pas l'option permettant de spécifier un compartiment ou un ID de sous-réseau, vérifiez que vous utilisez la version Par défaut la plus récente du document.

9. (Facultatif) Dans la section Balises, appliquez une ou plusieurs paires nom/valeur de clé de balise pour aider à identifier l'exécution, par exemple `Key=Purpose,Value=EC2Rescue`.
10. Sélectionnez Exécuter.

L'automatisation crée une AMI de sauvegarde dans le cadre du flux de travail. Toutes les autres ressources créées par le flux de travail d'automatisation sont automatiquement supprimées, mais cette AMI reste dans votre compte. L'AMI est nommée selon la convention suivante :

AMI de sauvegarde : AWSSupport-EC2Rescue :*`UnreachableInstanceId`*

Vous pouvez localiser cette AMI dans la console Amazon EC2 en recherchant l'ID d'exécution d'Automation.

Procédure : Réinitialiser les mots de passe et les clés SSH sur les instances EC2

Vous pouvez utiliser le document AWSSupport -ResetAccess pour réactiver automatiquement la génération du mot de passe administrateur local sur les instances EC2 pour et générer une nouvelle clé SSH sur les instances EC2 pour Linux.Windows Server Le document AWSSupport -ResetAccess est conçu pour effectuer une combinaison d'Systems Manageractions , d'AWS CloudFormationactions et de Lambda fonctions qui automatisent les étapes normalement requises pour réinitialiser le mot de passe administrateur local.

Vous pouvez utiliser Automation avec le document AWSSupport -ResetAccess pour résoudre les problèmes suivants :

Windows

Vous avez perdu la paire de clés EC2 : Pour résoudre ce problème, vous pouvez utiliser le document AWSSupport -ResetAccess pour créer une AMI activée par mot de passe à partir de votre instance actuelle, lancer une nouvelle instance à partir de l'AMI et sélectionner une paire de clés que vous possédez.

Vous avez perdu le mot de passe administrateur local : Pour résoudre ce problème, vous pouvez utiliser le document AWSSupport -ResetAccess pour générer un nouveau mot de passe que vous pouvez déchiffrer avec la paire de clés EC2 actuelle.

Linux

Vous avez perdu votre paire de clés EC2 ou vous avez configuré l'accès SSH à l'instance avec une clé que vous avez perdue : Pour résoudre ce problème, vous pouvez utiliser le document AWSSupport -ResetAccess pour créer une nouvelle clé SSH pour votre instance actuelle, ce qui vous permet de vous connecter à nouveau à l'instance.

> **Note**
>
> Si votre instance EC2 pour Windows Server est configurée pour Systems Manager , vous pouvez également réinitialiser votre mot de passe administrateur local en utilisant EC2Rescue et Fonctionnalité Exécuter la commande . Pour plus d'informations, consultez Utilisation EC2Rescue d' pour Windows Server avec Systems Manager Fonctionnalité Exécuter la commande dans le Amazon EC2 Guide de l'utilisateur pour les instances Windows .

Fonctionnement

Le dépannage d'une instance avec Automation et le document AWSSupport -ResetAccess fonctionne comme suit :

- Vous spécifiez l'ID de l'instance et exécutez le flux de travail d'automatisation.
- Le système crée un VPC temporaire, puis exécute une série de fonctions Lambda pour configurer le VPC.
- Le système identifie un sous-réseau pour votre VPC temporaire dans la même zone de disponibilité que votre instance d'origine.
- Le système lance une instance d'assistant SSM temporaire.
- Le système arrête l'instance originale et crée une sauvegarde. Ensuite, il rattache le volume racine original à l'instance d'assistant.
- Le système utilise Fonctionnalité Exécuter la commande pour s'exécuter EC2Rescue sur l'instance d'assistant. Sous Windows, EC2Rescue active la génération de mot de passe pour l'administrateur local en utilisant EC2Config ou EC2Launch sur le volume racine original attaché. Sous Linux, EC2Rescue génère et injecte une nouvelle clé SSH, et enregistre la clé privée (chiffrée) dans Parameter Store . Une fois terminé, EC2Rescue rattache le volume racine à l'instance d'origine.
- Le système crée une Amazon Machine Image (AMI) de votre instance, maintenant que la génération de mot de passe est activée. Vous pouvez utiliser cette AMI pour créer une instance EC2 et lui associer une nouvelle paire de clés le cas échéant.
- Le système redémarre votre instance originale et met fin à l'instance temporaire. Le système met également fin au VPC temporaire et aux fonctions Lambda créés au début de l'automatisation.
- Windows: Votre instance génère un nouveau mot de passe que vous pouvez décoder à partir de la Amazon EC2 console à l'aide de la paire de clés actuelle attribuée à l'instance.

  Linux : Vous pouvez accéder à l'instance via SSH en utilisant la clé SSH stockée dans Systems Manager Parameter Store sous la forme /ec2rl/openssh/*instance_id*/clé.

Avant de commencer

Avant d'exécuter l'automatisation suivante, veillez à exécuter les actions suivantes :

- Copiez l'ID de l'instance sur laquelle vous souhaitez réinitialiser le mot de passe administrateur. Vous spécifierez cet ID au cours de la procédure.
- Vous pouvez également recueillir l'ID d'un sous-réseau dans la même zone de disponibilité que votre instance inaccessible. LEC2Rescue'instance sera créée dans ce sous-réseau. Si vous ne spécifiez pas de sous-réseau, Automation crée un nouveau VPC temporaire dans votre compte AWS. Vérifiez que votre compte AWS possède au moins un VPC disponible. Par défaut, vous pouvez en créer cinq VPCs dans une région. Si vous en avez déjà créé cinq VPCs dans la région, l'automatisation échoue sans apporter de modifications à votre instance. Pour plus d'informations sur les Amazon VPC quotas, consultez VPC et sous-réseaux dans le Amazon VPC Guide de l'utilisateur .
- Vous pouvez éventuellement créer et spécifier un rôle AWS Identity and Access Management (IAM) pour l'automatisation. Si vous ne spécifiez pas ce rôle, Automation s'exécute dans le contexte de l'utilisateur qui a exécuté l'automatisation. Pour de plus amples informations sur la création de rôles pour l'automatisation, veuillez consulter Exécution d'une automatisation à l'aide d'un rôle IAM de service (p. 368).

Attribution d'autorisations AWSSupport -EC2Rescue pour effectuer des actions sur vos instances

EC2Rescuedoit être autorisé à effectuer une série d'actions sur vos instances au cours de l'exécution d'Automation. Ces actions appellent les services AWS Lambda, IAM et Amazon EC2 pour essayer en toute sécurité de corriger les problèmes liés à vos instances. Si vous avez des autorisations niveau administrateur dans votre compte AWS et/ou VPC, vous pouvez peut-être exécuter l'instance d'Automation sans configurer les autorisations, comme décrit dans cette section. Si vous ne possédez pas les autorisations niveau administrateur, donc vous, ou un administrateur, devez configurer les autorisations en utilisant l'une des options suivantes.

- Attribution des autorisations en utilisant les stratégies IAM (p. 663)
- Attribution des autorisations en utilisant un modèle AWS CloudFormation (p. 664)

Attribution des autorisations en utilisant les stratégies IAM

Vous pouvez attacher la stratégie IAM suivante à votre compte utilisateur, groupe ou rôle IAM en tant que stratégie en ligne, ou vous pouvez créer une nouvelle stratégie IAM gérée et la relier à votre compte utilisateur, groupe ou rôle. Pour plus d'informations au sujet de l'ajout d'une stratégie en ligne à votre compte utilisateur, groupe ou rôle, consultez la page Utilisation de stratégies en ligne. Pour plus d'informations sur la création d'une nouvelle stratégie gérée, consultez la page Utilisation de stratégies gérées.

> **Note**
>
> Si vous créez une nouvelle stratégie IAM gérée, vous devez également lui attacher la stratégie AmazonSSMAutomationRole gérée afin que vos instances puissent communiquer avec l'Systems ManagerAPI.

IAM Stratégie pour AWSSupport -ResetAccess

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Action": [
            "lambda:InvokeFunction",
            "lambda:DeleteFunction",
            "lambda:GetFunction"
         ],
         "Resource": "arn:aws:lambda:*:aws-account-id:function:AWSSupport-EC2Rescue-*",
         "Effect": "Allow"
      },
      {
         "Action": [
            "s3:GetObject",
            "s3:GetObjectVersion"
         ],
         "Resource": [
            "arn:aws:s3:::awssupport-ssm.*/*.template",
            "arn:aws:s3:::awssupport-ssm.*/*.zip"
         ],
         "Effect": "Allow"
      },
      {
         "Action": [
            "iam:CreateRole",
            "iam:CreateInstanceProfile",
            "iam:GetRole",
            "iam:GetInstanceProfile",
            "iam:PutRolePolicy",
            "iam:DetachRolePolicy",
```

```
            "iam:AttachRolePolicy",
            "iam:PassRole",
            "iam:AddRoleToInstanceProfile",
            "iam:RemoveRoleFromInstanceProfile",
            "iam:DeleteRole",
            "iam:DeleteRolePolicy",
            "iam:DeleteInstanceProfile"
         ],
         "Resource": [
            "arn:aws:iam::aws-account-id:role/AWSSupport-EC2Rescue-*",
            "arn:aws:iam::aws-account-id:instance-profile/AWSSupport-EC2Rescue-*"
         ],
         "Effect": "Allow"
      },
      {
         "Action": [
            "lambda:CreateFunction",
            "ec2:CreateVpc",
            "ec2:ModifyVpcAttribute",
            "ec2:DeleteVpc",
            "ec2:CreateInternetGateway",
            "ec2:AttachInternetGateway",
            "ec2:DetachInternetGateway",
            "ec2:DeleteInternetGateway",
            "ec2:CreateSubnet",
            "ec2:DeleteSubnet",
            "ec2:CreateRoute",
            "ec2:DeleteRoute",
            "ec2:CreateRouteTable",
            "ec2:AssociateRouteTable",
            "ec2:DisassociateRouteTable",
            "ec2:DeleteRouteTable",
            "ec2:CreateVpcEndpoint",
            "ec2:DeleteVpcEndpoints",
            "ec2:ModifyVpcEndpoint",
            "ec2:Describe*"
         ],
         "Resource": "*",
         "Effect": "Allow"
      }
   ]
}
```

Attribution des autorisations en utilisant un modèle AWS CloudFormation

AWS CloudFormation automatise le processus de création des rôles et stratégies IAM en utilisant un modèle préconfiguré. Utilisez la procédure suivante pour créer les IAM rôles et stratégies requis pour Automation à l'aide d' EC2Rescue .AWS CloudFormation

Pour créer les IAM rôles et stratégies requis pourEC2Rescue

1. Téléchargez `AWSSupport-EC2RescueRole.zip` et extrayez le fichier `AWSSupport-EC2RescueRole.json` pour l'enregistrer dans un répertoire de votre machine locale.
2. Si votre compte AWS est une partition spéciale, modifiez le modèle pour remplacer les valeurs de l'ARN par celles de votre partition.

   Par exemple, pour les Régions de Chine, remplacez toutes les occurrences de `arn:aws` par `arn:aws-cn`.
3. Connectez-vous à la AWS Management Console et ouvrez la console AWS CloudFormation à l'adresse https://console.aws.amazon.com/cloudformation.
4. Choisissez Créer une pile, Avec de nouvelles ressources (standard).
5. Sur la page Créer une pile, pour Prérequis - Préparer le modèle, choisissez Le modèle est prêt.

6. Dans Spécifier le modèle, choisissez Charger un modèle de fichier.
7. Sélectionnez Choisir le fichier, puis recherchez et sélectionnez le fichier `AWSSupport-EC2RescueRole.json` dans le répertoire où vous l'avez extrait.
8. Choisissez Nex t(Suivant).
9. Dans la page Spécifier les détails de la pile, pour le champ Nom de la pile, entrez un nom pour identifier cette pile, puis choisissez Suivant.
10. (Facultatif) Dans la zone Balises, appliquez une ou plusieurs paires nom/valeur de clé de balise à la pile.

    Les balises sont des métadonnées facultatives que vous affectez à une ressource. Les balises vous permettent de classer une ressource de différentes façons, par exemple, par objectif, par propriétaire ou par environnement. Par exemple, vous pouvez baliser une pile pour identifier le type de tâches qu'elle exécute, les types de cibles ou d'autres ressources concernées, et l'environnement dans lequel elle est exécutée.
11. Choisissez Suivant.
12. Sur la page Révision, passez en revue les détails de la pile, puis faites défiler vers le bas et choisissez l'option Je reconnais que AWS CloudFormation pourrait créer des ressources IAM.
13. AWS CloudFormation affiche l'état CREATE_IN_PROGRESS pendant quelques minutes. L'état passe à CREATE_COMPLETE après la création de la pile. Vous pouvez également choisir l'icône d'actualisation pour vérifier l'état du processus de création.
14. Dans la liste des piles, choisissez l'option en regard de la pile que vous venez de créer, puis sélectionnez l'onglet Sorties.
15. Copiez la Valeur. est l'ARN de l' AssumeRole . Vous spécifierez cet ARN lorsque vous exécuterez Automation.

Exécution d'Automation

La procédure suivante décrit comment exécuter le document AWSSupport -ResetAccess à l'aide de la AWS Systems Manager console.

> **Important**
>
> L'exécution d'Automation suivante arrête l'instance. L'arrêt de l'instance peut entraîner la perte de données sur des volumes de stockage d'instance attachés (le cas échéant). L'arrêt de l'instance peut aussi causer le changement de l'adresse IP publique, si aucune adresse IP Elastic n'est associée. Pour éviter ces changements de configuration, utilisez Fonctionnalité Exécuter la commande pour réinitialiser l'accès. Pour plus d'informations, consultez Utilisation EC2Rescue d' pour Windows Server avec Systems Manager Fonctionnalité Exécuter la commande dans le Amazon EC2 Guide de l'utilisateur pour les instances Windows .

Pour exécuter l'automatisation AWSSupport -ResetAccess

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Automatisation.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Automation.
3. Choisissez Execute automation (Exécuter l'automatisation).
4. Dans la section Document d'automatisation, choisissez M'appartenant ou appartenant à Amazon dans la liste.
5. Dans la liste des documents, choisissez le bouton dans la carte pour AWSSupport -ResetAccess, puis choisissez Next (Suivant).

6. Sur la page Exécuter le document d'automatisation, choisissez Exécution simple.
7. Dans la section Détails du document, vérifiez que l'option Version du document est définie sur la version par défaut la plus importante. Par exemple, $DEFAULT ou 3 (par défaut).
8. Dans la section Paramètres d'entrée, spécifiez les paramètres suivants :
   a. Pour InstanceID , spécifiez l'ID de l'instance inaccessible.
   b. Pour SubnetId , spécifiez un sous-réseau dans un VPC existant dans la même zone de disponibilité que l'instance que vous avez spécifiée. Par défaut, Systems Manager crée un VPC, mais vous pouvez spécifier un sous-réseau dans un VPC existant si vous le souhaitez.

      Note

      Si vous ne voyez pas l'option permettant de spécifier un ID de sous-réseau, vérifiez que vous utilisez la version Par défaut la plus récente du document.
   c. Pour EC2RescueInstanceType , spécifiez un type pour l'EC2Rescueinstance. Le type d'instance par défaut est `t2.small`.
   d. Pour AssumeRole , si vous avez créé des rôles pour cette automatisation à l'aide de la AWS CloudFormation procédure décrite précédemment dans cette rubrique, spécifiez l'AssumeRoleARN que vous avez noté dans la AWS CloudFormation console .
9. (Facultatif) Dans la section Balises, appliquez une ou plusieurs paires nom/valeur de clé de balise pour aider à identifier l'exécution, par exemple `Key=Purpose,Value=ResetAccess`.
10. Sélectionnez Exécuter.
11. Pour contrôler les progrès de l'exécution, choisissez le flux de travail d'automatisation en cours d'exécution, puis sélectionnez l'onglet Étapes. Lorsque l'exécution est terminée, choisissez l'onglet Descriptions, puis sélectionnez Afficher la sortie pour consulter les résultats. Pour consulter la sortie des étapes individuelles, choisissez l'onglet Étapes, puis sélectionnez Afficher les sorties en regard d'une étape.

L'automatisation crée une AMI de sauvegarde et une AMI activée par mot de passe dans le cadre du flux de travail. Toutes les autres ressources créées par le flux de travail Automation sont automatiquement supprimées, mais elles AMIs restent dans votre compte. AMIsLes sont nommés selon les conventions suivantes :

- AMI de sauvegarde : `AWSSupport-EC2Rescue:`*`InstanceId`*
- AMI activée par mot de passe : AWSSupport-EC2Rescue : AMI activée par mot de passe à partir d' *`InstanceId`*

Vous pouvez les localiser AMIs en recherchant l'ID d'exécution d'Automation.

Pour Linux, la nouvelle clé privée SSH pour votre instance est enregistrée, chiffrée, dans Parameter Store. Le nom du paramètre est /ec2rl/openssh/***`instance_id`***/clé.

Procédure : Utilisation des transformateurs en entrée avec Automation

Cette procédure Systems Manager Automation montre comment utiliser la fonction de transformateur en entrée de Amazon EventBridge pour extraire l'`instance-id` d'une instance EC2 à partir d'un événement de changement d'état d'instance. Nous utilisons le transformateur d'entrée pour transmettre ces données à la cible du document d'automatisation `AWS-CreateImage` Systems Manager comme paramètre d'entrée `InstanceId`. La règle est déclenchée lorsqu'une instance passe à l'état `stopped`.

Pour plus d'informations sur l'utilisation des transformateurs d'entrée, consultez le didacticiel : Utilisez le transformateur d'entrée pour personnaliser ce qui est transmis à la cible de l'événement dans le Guide de l'utilisateur Amazon EventBridge .

Avant de commencer

Vérifiez que vous avez ajouté les autorisations et la stratégie d'approbation requises pour EventBridge à votre rôle de service d'automatisation Systems Manager. Pour plus d'informations, consultez Présentation de la gestion des autorisations d'accès à vos EventBridge ressources dans le Guide de l'utilisateur Amazon EventBridge .

Pour utiliser des transformateurs d'entrée avec l'automatisation

1. Ouvrez la console Amazon EventBridge à l'adresse https://console.aws.amazon.com/events/.
2. Dans le volet de navigation, choisissez Rules, puis Create rule.

   -ou-

   Si la page Amazon EventBridge d'accueil s'ouvre d'abord, choisissez Create rule (Créer une règle).
3. Saisissez un nom et une description pour la règle.

   Une règle ne peut pas avoir le même nom qu'une autre règle de la même région et sur le même bus d'événement.
4. Pour Define pattern (Définir un modèle), procédez comme suit :
   a. Choisissez Event pattern.
   b. Pour Modèle de correspondance d'événement, choisissez Modèle prédéfini par un service.
   c. Pour Modèle de correspondance d'événement, choisissez Modèle prédéfini par un service.
   d. Pour Fournisseur de service, choisissez AWS.
   e. Pour Service name (Nom du service), choisissez EC2.
   f. Pour Event Type (Type d'événement), choisissez EC2 Instance State-change Notification (Notification de changement d'état de l'instance EC2).
   g. Choisissez État(s) spécifique(s) et arrêté dans la liste déroulante.
   h. Choisissez N'importe quelle instance.
5. Pour Select event bus (Sélectionner un bus d'événement), choisissez le bus d'événement que vous souhaitez associer à cette règle. Si vous souhaitez que cette règle se déclenche sur les événements correspondants provenant de votre propre compte AWS, sélectionnez AWS default event bus (Bus d'événement par défaut AWS). Lorsqu'un service AWS de votre compte émet un événement, il accède toujours au bus d'événement par défaut de votre compte.
6. Pour Target (Cible), choisissez SSM Automation (Automatisation SSM).
7. Pour Document, choisissez AWS-CreateImage.
8. Développez Configurer les paramètres d'automatisation et choisissez Transformateur d'entrée.
9. Dans la première zone sous Transformateur d'entrée, entrez **`{"instance":"$.detail.instance-id"}`** .
10. Dans la deuxième zone, entrez **`{"InstanceId":[<instance>]}`** .
11. Choisissez Utiliser le rôle existant et choisissez votre rôle de service d'automatisation.
12. (Facultatif) Entrez une ou plusieurs balises pour la règle. Pour de plus amples informations, veuillez consulter Balisage de vos ressources Amazon EventBridge dans le Guide de l'utilisateur Amazon EventBridge.
13. Sélectionnez Create.

Procédure : Utilisation d'Automation avec Jenkins

Si votre organisation utilise le logiciel Jenkins dans un pipeline CI/CD, vous pouvez ajouter Automation comme étape après génération pour préinstaller des versions d'application dans des Amazon Machines

Images (AMI). Vous pouvez aussi utiliser la fonction de planification de Jenkins pour appeler Automation et créer votre propre rythme d'application de correctifs de système d'exploitation.

L'exemple ci-dessous montre comment appeler un serveur Jenkins qui s'exécute sur site ou dans Amazon EC2. Pour l'authentification, le serveur Jenkins utilise les informations d'identification AWS en fonction d'un utilisateur AWS Identity and Access Management (IAM) que vous créez dans l'exemple. Si votre serveur Jenkins s'exécute dans Amazon EC2, vous pouvez également l'authentifier à l'aide d'un rôle de profil d'instance IAM.

> **Note**
>
> Veillez à respecter les bonnes pratiques de sécurité de Jenkins quand vous configurez votre instance.

Avant de commencer

Effectuez les tâches suivantes avant de configurer Automation avec Jenkins.

- Complétez l'exemple Procédure : Simplification de l'application de correctifs à l'AMI avec Automation, AWS Lambda et Parameter Store (p. 645). L'exemple suivant utilise le document UpdateMyLatestWindowsAmi d'automatisation créé dans cet exemple.
- Configurez des rôles IAM pour Automation. Systems Manager nécessite un rôle de profil d'instance et un ARN de rôle de service pour traiter les flux de travail d'automatisation. Pour plus d'informations, consultez la section Configuration d'Automation (p. 308).
- Après avoir configuré des rôles IAM pour Automation, utilisez la procédure suivante afin de créer un compte utilisateur IAM pour votre serveur Jenkins. Les flux de travail Automation utilisent la clé d'accès et la clé secrète du compte utilisateur IAM pour authentifier le serveur Jenkins lors de l'exécution.

Pour créer un compte utilisateur pour le serveur Jenkins

1. Connectez-vous à la AWS Management Console et ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation, choisissez Stratégies, puis Créer une stratégie.
3. Choisissez l'onglet JSON.
4. Remplacez le contenu par défaut par la commande suivante : Veillez à remplacer *us-west-2* et *123456789012* avec la région et le compte que vous souhaitez utiliser.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ssm:StartAutomationExecution",
            "Resource": [
                "arn:aws:ssm:us-west-2:123456789012:document/UpdateMyLatestWindowsAmi",
                "arn:aws:ssm:us-west-2:123456789012:automation-definition/
UpdateMyLatestWindowsAmi:$DEFAULT"
            ]
        }
    ]
}
```

5. Choisissez Examiner une stratégie.
6. Sur la page Examiner une stratégie, dans le champ Nom, saisissez un nom pour la stratégie en ligne, tel que **`JenkinsPolicy`**.
7. Choisissez Create policy (Créer une stratégie).
8. Dans le panneau de navigation, sélectionnez Users (Utilisateurs).

9. Choisissez Ajouter un utilisateur.
10. Dans la section Set user details (Définir les détails de l'utilisateur), spécifiez un nom d'utilisateur (par exemple, Jenkins).
11. Dans la section Select AWS access type (Sélectionner le type d'accès AWS), choisissez Programmatic Access (Accès par programme).
12. Choisissez Suivant : Autorisations.
13. Dans la section Définir les autorisations pour, choisissez Attacher directement les stratégies existantes.
14. Dans le champ de filtre, tapez le nom de la stratégie que vous avez créée précédemment.
15. Cochez la case en regard de la stratégie, puis choisissez Suivant : Balises.
16. (Facultatif) Ajoutez une ou plusieurs paires clé-valeur de balise pour organiser, suivre ou contrôler l'accès pour cet utilisateur, puis choisissez Suivant : Passez en revue.
17. Vérifiez les détails, puis choisissez Create (Créer).
18. Copiez la clé d'accès et la clé secrète dans un fichier texte. Vous spécifierez ces informations d'identification dans la procédure suivante.

Utilisez la procédure suivante pour configurer l'AWS CLI sur votre serveur Jenkins.

Pour configurer le serveur Jenkins pour Automation

1. Connectez-vous à votre serveur Jenkins sur le port 8080 à l'aide de votre navigateur préféré pour accéder à l'interface de gestion.
2. Saisissez le mot de passe trouvé dans `/var/lib/jenkins/secrets/initialAdminPassword` . Pour afficher votre mot de passe, exécutez la commande suivante :

   ```
   sudo cat /var/lib/jenkins/secrets/initialAdminPassword
   ```

3. Le script d'installation Jenkins vous dirige vers la page Personnaliser Jenkins. Sélectionnez Installer les plug-ins suggérés.
4. Une fois l'installation terminée, choisissez Informations d'identification de l'administrateur, sélectionnez Enregistrer les informations d'identification, puis sélectionnez Démarrer à l'aide de Jenkins.
5. Dans le volet de navigation de gauche, choisissez Manage Jenkins (Gérer Jenkins), puis Manage Plugins (Gérer les plug-ins).
6. Choisissez l'onglet Disponible, puis entrez **`Amazon EC2 plugin`** .
7. Cochez la case correspondant **`Amazon EC2 plugin`** à , puis sélectionnez Installer sans redémarrer.
8. Une fois l'installation terminée, sélectionnez Aller en haut de la page.
9. Choisissez Manage Jenkins (Gérer Jenkins), puis Configure System (Configurer le système).
10. Dans la section Cloud, sélectionnez Ajouter un nouveau cloud, puis choisissez Amazon EC2.
11. Saisissez vos informations dans les champs restants. Notez que vous devez entrer vos informations d'identification AWS dans le champ Ajouter les informations d'identification.

Utilisez la procédure suivante afin de configurer votre projet Jenkins pour appeler Automation.

Pour configurer votre serveur Jenkins afin d'appeler Automation

1. Ouvrez la console Jenkins dans un navigateur Web.
2. Choisissez le projet que vous voulez configurer avec Automation, puis sélectionnez Configure.
3. Dans l'onglet Build, choisissez Add Build Step.
4. Choisissez Execute shell ou Execute Windows batch command (en fonction de votre système d'exploitation).

5. Dans la zone Command (Commande), exécutez une commande de l'AWS CLI comme suit :

```
aws --region the AWS Region of your source AMI ssm start-automation-execution --
document-name your document name --parameters parameters for the document
```

L'exemple de commande suivant utilise le UpdateMyLatestWindowsAmi document et le Systems Manager paramètre `latestAmi` créé dans Procédure : Simplification de l'application de correctifs à l'AMI avec Automation, AWS Lambda et Parameter Store (p. 645) :

```
aws --region region-id ssm start-automation-execution \
    --document-name UpdateMyLatestWindowsAmi \
    --parameters \
        "sourceAMIid='{{ssm:latestAmi}}'"
```

Dans Jenkins, la commande ressemble à l'exemple dans la capture d'écran suivante.

6. Dans le projet Jenkins, choisissez Build Now. Jenkins retourne une sortie similaire à l'exemple suivant.

Procédure : Utilisation de Document Builder pour créer un document Automation personnalisé

La procédure suivante montre comment utiliser Document Builder dans la console Systems Manager Automation pour créer un document Automation personnalisé, puis l'exécuter.

La première étape du document Automation que vous créez exécute un script pour lancer une instance Amazon Elastic Compute Cloud (EC2). La deuxième étape exécute un autre script pour surveiller la vérification de l'état de l'instance à remplacer par `ok` . Ensuite, un état général de `Success` est signalé pour l'exécution de l'automatisation.

Avant de commencer

Avant de commencer cette procédure, exécutez les étapes suivantes :

- Vérifiez que vous disposez de privilèges d'administrateur ou que vous avez obtenu les autorisations adéquates d'accès à Systems Manager dans AWS Identity and Access Management (IAM).

  Pour plus d'informations, consultez la section Vérification de l'accès des utilisateurs aux flux de travail Automation (p. 308).
- Vérifiez que vous disposez d'un rôle de service IAM pour Automation (également appelé rôle de responsable) dans votre compte AWS. Ce rôle est obligatoire car cette procédure utilise l'action aws:executeScript.

  Pour plus d'informations sur la création de ce rôle, consultez Configuration d'un accès à un rôle de service (rôle de responsable) pour les flux de travail Automation (p. 308).

  Pour plus d'informations sur le rôle de service IAM requis pour exécuter aws:executeScript, consultez Autorisations pour exécuter des exécutions Automation (p. 451).
- Vérifiez que vous avez l'autorisation de lancer des instances EC2.

  Pour plus d'informations, consultez IAM et Amazon EC2 dans le Amazon EC2 Guide de l'utilisateur pour les instances Linux.

Rubriques

- Étape 1 : Créer le document Automation personnalisé (p. 671)
- Étape 2 : Exécuter le document Automation personnalisé (p. 676)

Étape 1 : Créer le document Automation personnalisé

Utilisez la procédure suivante pour créer un document Automation personnalisé qui lance une instance EC2 et attend la modification de la vérification d'état de l'instance en `ok`.

> **Tip**
>
> Si vous copiez et collez des valeurs de cette procédure dans Document Builder, par exemple, les noms de paramètres et les noms de gestionnaire, assurez-vous de supprimer les espaces de début ou de fin ajoutés à la valeur de texte que vous entrez.

Pour créer un document Automation personnalisé à l'aide de Document Builder

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Documents dans le panneau de navigation.
3. Choisissez Create automation (Créer une automatisation).
4. Pour Name, tapez ce nom descriptif pour le document : **`LaunchInstanceAndCheckStatus`**.
5. (Facultatif) Pour Document description (Description du document), remplacez le texte par défaut par une description pour ce document, à l'aide de Markdown. Voici un exemple pour .

```
##Title: LaunchInstanceAndCheckState
-----
**Purpose**: This Automation document first launches an EC2 instance using the AMI ID
 provided in the parameter ```imageId```. The second step of this document continuously
 checks the instance status check value for the launched instance until the status
 ```ok``` is returned.
```
```

```
##Parameters:
-----
Name | Type | Description | Default Value
------------- | ------------- | ------------- | -------------
assumeRole | String | (Optional) The ARN of the role that allows Automation to perform
 the actions on your behalf. | -
imageId  | String | (Optional) The AMI ID to use for launching the instance. The
 default value uses the latest Amazon Linux AMI ID available. | {{ ssm:/aws/service/
ami-amazon-linux-latest/amzn-ami-hvm-x86_64-gp2 }}
```

6. Pour Assume role (Rôle responsable), entrez l'ARN du rôle de service pour Automation (Rôle responsable) pour l'exécution de l'automatisation, au format IAM .**`arn:aws:iam::111122223333:role/AutomationServiceRole`** Remplacez votre ID de compte AWS par 111122223333 .

   Le rôle que vous spécifiez est utilisé pour fournir les autorisations nécessaires pour démarrer l'exécution de l'automatisation.

   > **Important**
   >
   > Pour les documents Automation qui ne sont pas la propriété d'Amazon et qui utilisent l'action `aws:executeScript`, un rôle doit être spécifié. Pour plus d'informations, consultez la section Autorisations pour exécuter des exécutions Automation (p. 451).

7. Développez Input parameters (Paramètres d'entrée) et procédez comme suit.

   1. Dans Parameter name (Nom du paramètre), entrez **`imageId`**.
   2. Dans le champ Type, choisissez **`String`**.
   3. Pour Obligatoire, choisissez `No`.
   4. Pour Default value (Valeur par défaut), entrez ce qui suit.

      ```
      {{ ssm:/aws/service/ami-amazon-linux-latest/amzn-ami-hvm-x86_64-gp2 }}
      ```

      > **Note**
      >
      > Cette valeur lance une instance EC2 à l'aide du dernier ID Amazon Machine Image (AMI) Amazon Linux. Si vous souhaitez utiliser une autre AMI, remplacez la valeur par l'ID de votre AMI.

   5. Pour Description, entrez ce qui suit.

      ```
      (Optional) The AMI ID to use for launching the instance. The default value uses the
       latest released Amazon Linux AMI ID.
      ```

8. Choisissez Add a parameter (Ajouter un paramètre) pour créer le second paramètre **`tagValue`**, puis entrez ce qui suit.

   1. Dans Parameter name (Nom du paramètre), entrez **`tagValue`**.
   2. Dans le champ Type, choisissez **`String`**.
   3. Pour Obligatoire, choisissez `No`.
   4. Pour Valeur par défaut, entrez **`LaunchedBySsmAutomation`** . Cela ajoute la valeur de paire de clés de balise `Name:LaunchedBySsmAutomation` à l'instance.
   5. Pour Description, entrez ce qui suit.

      ```
      (Optional) The tag value to add to the instance. The default value is
       LaunchedBySsmAutomation.
      ```

9. Choisissez Add a parameter (Ajouter un paramètre) pour créer le troisième paramètre **`instanceType`**, puis entrez les informations suivantes.

1. Dans Parameter name (Nom du paramètre), entrez **`instanceType`**.
2. Dans le champ Type, choisissez **`String`**.
3. Pour Obligatoire, choisissez `No`.
4. Pour Default value (Valeur par défaut), entrez **`t2.micro`**.
5. Pour Parameter description (Description du paramètre), entrez ce qui suit.

```
(Optional) The instance type to use for the instance. The default value is t2.micro.
```

10. Développez Target type (Type de cible) et choisissez **`"/"`**.
11. (Facultatif) Développez les Document tags (Balises de document) pour appliquer des balises de ressources à votre document Automation. Pour la Tag key (Clé de balise), entrez **`Purpose`**, et pour Tag value (Valeur de balise), entrez **`LaunchInstanceAndCheckState`**.
12. Dans la section Step 1 (Étape 1) procédez comme suit.

    1. Pour Step name (Nom de l'étape), entrez ce nom d'étape descriptif pour la première étape du flux de travail d'automatisation : **`LaunchEc2Instance`**.
    2. Pour Action type (Type d'action), sélectionnez Run a script (Exécuter un script= (**`aws:executeScript`**).
    3. Dans Description, entrez une description de l'étape d'automatisation, telle que la suivante.

```
**About This Step**

This step first launches an EC2 instance using the ```aws:executeScript``` action and
 the provided script.
```

    4. Développez Inputs (Entrées).
    5. Dans Exécution, choisissez le langage d'exécution à utiliser pour exécuter le script fourni.
    6. Pour Handler (Gestionnaire), entrez .**`launch_instance`** Il s'agit du nom de la fonction déclaré dans le script suivant.

       Note

       Ceci n'est pas obligatoire pour PowerShell .

    7. Pour Script, remplacez le contenu par défaut par ce qui suit. Veillez à faire correspondre le script avec la valeur d'exécution correspondante.

       Python

```
def launch_instance(events, context):
  import boto3
  ec2 = boto3.client('ec2')

  image_id = events['image_id']
  tag_value = events['tag_value']
  instance_type = events['instance_type']

  tag_config = {'ResourceType': 'instance', 'Tags': [{'Key':'Name',
 'Value':tag_value}]}

  res = ec2.run_instances(ImageId=image_id, InstanceType=instance_type,
 MaxCount=1, MinCount=1, TagSpecifications=[tag_config])

  instance_id = res['Instances'][0]['InstanceId']

  print('[INFO] 1 EC2 instance is successfully launched', instance_id)

  return { 'InstanceId' : instance_id }
```

PowerShell

```
Install-Module AWS.Tools.EC2 -Force
Import-Module AWS.Tools.EC2

$payload = $env:InputPayload | ConvertFrom-Json

$imageid = $payload.image_id

$tagvalue = $payload.tag_value

$instanceType = $payload.instance_type

$type = New-Object Amazon.EC2.InstanceType -ArgumentList $instanceType

$resource = New-Object Amazon.EC2.ResourceType -ArgumentList 'instance'

$tag = @{Key='Name';Value=$tagValue}

$tagSpecs = New-Object Amazon.EC2.Model.TagSpecification

$tagSpecs.ResourceType = $resource

$tagSpecs.Tags.Add($tag)

$res = New-EC2Instance -ImageId $imageId -MinCount 1 -MaxCount 1 -InstanceType
 $type -TagSpecification $tagSpecs

return @{'InstanceId'=$res.Instances.InstanceId}
```

8. Développer Additional inputs (Entrées supplémentaires).
9. Pour Input name (Nom d'entrée), choisissez InputPayload . Pour Valeur d'entrée, entrez les données YAML suivantes.

```
image_id: "{{ imageId }}"
tag_value: "{{ tagValue }}"
instance_type: "{{ instanceType }}"
```

13. Développez Outputs (Sorties) et procédez comme suit :
    - Pour Nom, saisissez **`payload`**.
    - Pour Selector (Sélecteur), entrez **`$.Payload`**.
    - Pour Type, choisissez `StringMap`.
14. Choisissez Add step (Ajouter étape) pour ajouter une deuxième étape au document Automation. La deuxième étape interroge l'état de l'instance lancée à l'étape 1 et attend que l'état renvoyé soit `ok`.
15. Dans la section Step 2 (Étape 2) procédez comme suit.
    1. Pour Step name (Nom de l'étape), entrez ce nom descriptif pour la deuxième étape du flux de travail d'automatisation : **`WaitForInstanceStatusOk`**.
    2. Pour Action type (Type d'action), sélectionnez Run a script (Exécuter un script= (**`aws:executeScript`**).
    3. Dans Description, entrez une description de l'étape d'automatisation, telle que la suivante.

```
**About This Step**

The script continuously polls the instance status check value for the instance
 launched in Step 1 until the ```ok``` status is returned.
```

4. Pour Exécution, choisissez le langage d'exécution à utiliser pour exécuter le script fourni.
5. Pour Handler (Gestionnaire), entrez **.poll_instance** Il s'agit du nom de la fonction déclaré dans le script suivant.

   Note

   Ceci n'est pas obligatoire pour PowerShell .

6. Pour Script, remplacez le contenu par défaut par ce qui suit. Veillez à faire correspondre le script avec la valeur d'exécution correspondante.

   Python

```
def poll_instance(events, context):
  import boto3
  import time

  ec2 = boto3.client('ec2')

  instance_id = events['InstanceId']

  print('[INFO] Waiting for instance status check to report ok', instance_id)

  instance_status = "null"

  while True:
    res = ec2.describe_instance_status(InstanceIds=[instance_id])

    if len(res['InstanceStatuses']) == 0:
      print("Instance status information is not available yet")
      time.sleep(5)
      continue

    instance_status = res['InstanceStatuses'][0]['InstanceStatus']['Status']

    print('[INFO] Polling to get status of the instance', instance_status)

    if instance_status == 'ok':
      break

    time.sleep(10)

  return {'Status': instance_status, 'InstanceId': instance_id}
```

   PowerShell

```
Install-Module AWS.Tools.EC2 -Force

$inputPayload = $env:InputPayload | ConvertFrom-Json

$instanceId = $inputPayload.payload.InstanceId

$status = Get-EC2InstanceStatus -InstanceId $instanceId

while ($status.Status.Status -ne 'ok'){
   Write-Host 'Polling get status of the instance', $instanceId

   Start-Sleep -Seconds 5

   $status = Get-EC2InstanceStatus -InstanceId $instanceId
}
```

```
return @{Status = $status.Status.Status; InstanceId = $instanceId}
```

7. Développer Additional inputs (Entrées supplémentaires).
8. Pour Input name (Nom d'entrée), choisissez InputPayload . Pour la valeur d'entrée, entrez ce qui suit :

```
{{ LaunchEc2Instance.payload }}
```

16. Choisissez Create automation (Créer une automatisation) pour enregistrer le document.

Étape 2 : Exécuter le document Automation personnalisé

Suivez la procédure suivante pour exécuter le document Automation personnalisé créé à l'étape 1. Le document Automation personnalisé lance une instance EC2 et attend que la vérification de l'instance passe à l'état `ok`.

Pour exécuter le document Automation personnalisé

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Automation (Automatisation), puis Execute automation (Exécuter l'automatisation).
3. Dans la liste Automation document (Document Automation) choisissez l'onglet Owned by me (M'appartenant) puis cliquez sur le bouton en regard du document Automation personnalisé que vous avez créé **`LaunchInstanceAndCheckStatus`**.
4. Dans la section Document details (Détails du document), pour Document version (Version du document), vérifiez que l'option Default version at runtime (Version par défaut à l'exécution) est sélectionnée.
5. Choisissez Nex t(Suivant).
6. En haut de la page du Execute automation document (Exécuter le document Automation) vérifiez que l'option Simple execution (Exécution simple) est sélectionnée.
7. Sélectionnez Exécuter.
8. Une fois les deux étapes du flux de travail d'automatisation terminées, dans la zone Executed steps (Étapes exécutées) choisissez l'ID d'étape d'une étape pour afficher les détails des étapes, y compris toute sortie d'étape.

   Note

   Le renvoi de l'état `ok` peut prendre plusieurs minutes.
9. (Facultatif) Sauf si vous prévoyez d'utiliser l'instance EC2 créée par cette procédure à d'autres fins, vous pouvez résilier l'instance. Pour plus d'informations, consultez Terminate Your Instance (Résilier votre instance) dans le manuel Amazon EC2 Guide de l'utilisateur pour les instances Linux.

   Vous pouvez identifier l'instance par le nom avec LaunchedBySsmAutomation lequel vous l'avez balisée dans Étape 1 : Créer le document Automation personnalisé (p. 671) .

Dépannage de Systems Manager Automation

Utilisez les informations suivantes pour dépanner les problèmes du service Automation. La rubrique inclut des tâches spécifiques pour résoudre des problèmes selon les messages d'erreur d'Automation.

Rubriques

- Erreurs d'automatisation courantes (p. 677)
- L'exécution d'Automation n'a pas pu démarrer (p. 677)

- L'exécution a démarré, mais le statut est Failed (Échec) (p. 678)
- L'exécution a démarré, mais a expiré (p. 680)

Erreurs d'automatisation courantes

Cette section inclut des informations sur les erreurs d'Automation courantes.

VPC non défini 400

Par défaut, lorsqu'Automation exécute les documents `AWS-UpdateLinuxAmi` ou `AWS-UpdateWindowsAmi`, le système crée une instance temporaire dans le VPC par défaut (172.30.0.0/16). Si vous avez supprimé le VPC par défaut, vous recevrez l'erreur suivante :

```
VPC not defined 400
```

Pour résoudre ce problème, vous devez spécifier une valeur pour le paramètre d'entrée `SubnetId`.

L'exécution d'Automation n'a pas pu démarrer

Une exécution d'Automation peut échouer à cause d'une erreur d'accès refusé ou d'une erreur de rôle de responsable non valide si vous n'avez pas correctement configuré les utilisateurs, rôles et stratégies IAM pour Automation.

Accès refusé

Les exemples suivants décrivent les situations lorsqu'une exécution d'Automation a échoué lors du démarrage, avec une erreur d'accès refusé.

Accès refusé à l'API Systems Manager

Message d'erreur : `User: user arn is not authorized to perform: ssm:StartAutomationExecution on resource: document arn (Service: AWSSimpleSystemsManagement; Status Code: 400; Error Code: AccessDeniedException; Request ID: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)`

- Cause possible 1 : LIAM'utilisateur qui tente de lancer l'exécution d'Automation n'a pas l'autorisation d'appeler l'`StartAutomationExecution`API. Pour résoudre ce problème, attachez la stratégie IAM requise au compte utilisateur qui a été utilisé pour lancer l'exécution. Pour plus d'informations, consultez la section Tâche 3 : Configuration de l'accès utilisateur à Automation (p. 313).
- Cause possible 2 : LIAM'utilisateur qui tente de lancer l'exécution d'Automation a l'autorisation d'appeler l`StartAutomationExecution`'API, mais n'a pas l'autorisation d'appeler l'API à l'aide du document Automation spécifique. Pour résoudre ce problème, attachez la stratégie IAM requise au compte utilisateur qui a été utilisé pour lancer l'exécution. Pour plus d'informations, consultez la section Tâche 3 : Configuration de l'accès utilisateur à Automation (p. 313).

Accès refusé en raison d'autorisations manquantesPassRole

Message d'erreur : `User: user arn is not authorized to perform: iam:PassRole on resource: automation assume role arn (Service: AWSSimpleSystemsManagement; Status Code: 400; Error Code: AccessDeniedException; Request ID: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)`

LIAM'utilisateur qui tente de lancer l'exécution d'Automation nPassRole'a pas l'autorisation pour le rôle de responsable. Pour résoudre ce problème, attachez la stratégie iam:PassRole au rôle de l'utilisateur IAM tentant de lancer l'exécution d'Automation. Pour plus d'informations, consultez la section Tâche 2 : Attachement de la stratégie iam:PassRole à votre rôle Automation (p. 312).

Rôle de responsable non valide

Lorsque vous exécutez Automation, un rôle de responsable (assume) est fourni dans le document ou transmis en tant que valeur de paramètre pour le document. Différents types d'erreurs peuvent survenir si le rôle de responsable n'est pas correctement spécifié ou configuré.

Rôle de responsable incorrect

Message d'erreur : `The format of the supplied assume role ARN is invalid.`Le rôle de responsable n'est pas correctement formaté. Pour résoudre ce problème, vérifiez qu'un rôle de responsable valide est spécifié dans votre document Automation ou en tant que paramètre d'exécution lorsque vous exécutez Automation.

Le rôle de responsable ne peut être assumé

Message d'erreur : `The defined assume role is unable to be assumed. (Service: AWSSimpleSystemsManagement; Status Code: 400; Error Code: InvalidAutomationExecutionParametersException; Request ID: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)`

- Cause possible 1 : Le rôle de responsable n'existe pas. Pour résoudre ce problème, créez le rôle. Pour plus d'informations, consultez la section the section called “Configuration d'Automation” (p. 308). Des détails spécifiques pour créer ce rôle sont décrits dans la rubrique suivante, Tâche 1 : Créer un rôle de service pour Automation (p. 310).
- Cause possible 2 : Le rôle assume n'a pas de relation d'approbation avec le Systems Manager service. Pour résoudre ce problème, créez la relation d'approbation. Pour de plus amples informations, veuillez consulter Je ne peux pas assumer un rôle dans le IAM Guide de l'utilisateur.

L'exécution a démarré, mais le statut est Failed (Échec)

Échecs spécifiques à l'action

Les documents Automation contiennent des étapes et les étapes sont exécutées dans l'ordre. Chaque étape appelle un ou plusieurs AWS services APIs . APIsdétermine les entrées, le comportement et les sorties de l'étape. Il existe plusieurs situations dans lesquelles une erreur peut causer l'échec d'une étape. Les messages d'échec indiquent quand et où une erreur s'est produite.

Pour afficher un message d'échec dans la console EC2, sélectionnez le lien View Outputs de l'étape échouée. Pour afficher un message d'échec de l'AWS CLI, appelez `get-automation-execution` et recherchez l'attribut `FailureMessage` dans une `StepExecution` échouée.

Dans les exemples suivants, une étape associée à l'action `aws:runInstance` a échoué. Chaque exemple décrit un type d'erreur différent.

Image manquante

Message d'erreur : `Automation Step Execution fails when it is launching the instance(s). Get Exception from RunInstances API of ec2 Service. Exception Message from RunInstances API: [The image id '[ami id]' does not exist (Service: AmazonEC2; Status Code: 400; Error Code: InvalidAMIID.NotFound; Request ID: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)]. Please refer to Automation Service Troubleshooting Guide for more diagnosis details.`

L'action `aws:runInstances` a reçu une entrée pour une `ImageId` qui n'existe pas. Pour résoudre ce problème, mettez à jour le document d'Automation ou les valeurs de paramètre avec l'ID d'AMI correct.

Le rôle de responsable ne dispose pas des autorisations suffisantes

Message d'erreur : `Automation Step Execution fails when it is launching the instance(s). Get Exception from RunInstances API of ec2 Service. Exception`

```
Message from RunInstances API: [You are not authorized to perform this
operation. Encoded authorization failure message: xxxxxxx (Service: AmazonEC2;
Status Code: 403; Error Code: UnauthorizedOperation; Request ID: xxxxxxxx-xxxx-
xxxx-xxxx-xxxxxxxxxxxx)]. Please refer to Automation Service Troubleshooting
Guide for more diagnosis details.
```

Le rôle assume ne dispose pas des autorisations suffisantes pour appeler l'`RunInstances`API sur des instances EC2. Pour résoudre ce problème, attachez une stratégie IAM au rôle assume qui a l'autorisation d'appeler l'API `RunInstances`. Pour de plus amples informations, veuillez consulter Méthode 2 : Utiliser IAM pour configurer des rôles pour Automation (p. 310).

État inattendu

Message d'erreur : `Step fails when it is verifying launched instance(s) are ready to be used. Instance i-xxxxxxxxx entered unexpected state: shutting-down. Please refer to Automation Service Troubleshooting Guide for more diagnosis details.`

- Cause possible 1 : Il y a un problème avec l'instance ou le Amazon EC2 service. Pour résoudre ce problème, connectez-vous à l'instance ou vérifiez le journal système de l'instance pour comprendre ce qui a déclenché l'arrêt de l'instance.
- Cause possible 2 : Le script de données utilisateur spécifié pour l'`aws:runInstances`action a un problème ou une syntaxe incorrecte. Vérifiez la syntaxe du script des données utilisateur. Vérifiez aussi que le script des données utilisateur n'arrête pas l'instance ou n'appelle pas d'autres scripts susceptibles d'arrêter l'instance.

Référence des échecs spécifiques à l'action

Lorsqu'une étape échoue, le message peut indiquer quel service a été appelé au moment de l'échec. Le tableau suivant répertorie les services appelés par chaque action. Le tableau fournit aussi des liens vers des informations sur chaque service.

Action	Service(s) AWS appelé(s) par cette action	Pour obtenir des informations sur ce service	Contenu de dépannage
aws:runInstances	Amazon EC2	Amazon EC2 Guide de l'utilisateur pour les instances Linux	Dépannage des instances EC2
aws:changeInstanceState	Amazon EC2	Amazon EC2 Guide de l'utilisateur pour les instances Linux	Dépannage des instances EC2
aws:runCommand	Systems Manager	AWS Systems Manager Fonctionnalité Exécuter la commande (p. 941)	Dépannage de la fonctionnalité Exécuter la commande de Systems Manager (p. 986)
aws:createImage	Amazon EC2	Amazon Machine Images	
aws:createStack	AWS CloudFormation	AWS CloudFormation Guide de l'utilisateur	Dépannage AWS CloudFormation
aws:deleteStack	AWS CloudFormation	AWS CloudFormation Guide de l'utilisateur	Dépannage AWS CloudFormation

Action	Service(s) AWS appelé(s) par cette action	Pour obtenir des informations sur ce service	Contenu de dépannage
aws:deleteImage	Amazon EC2	Amazon Machine Images	
aws:copyImage	Amazon EC2	Amazon Machine Images	
aws:createTag	Amazon EC2, Systems Manager	Ressources et balises EC2	
aws:invokeLambdaFunction	AWS Lambda	AWS Lambda Developer Guide	Dépannage Lambda

Erreur interne du service Automation

Message d'erreur : `Internal Server Error. Please refer to Automation Service Troubleshooting Guide for more diagnosis details.`

Un problème du service Automation empêche le document Automation spécifié de s'exécuter correctement. Pour résoudre ce problème, contactez AWS Support. Fournissez l'ID d'exécution et l'ID client, si disponible.

L'exécution a démarré, mais a expiré

Message d'erreur : `Step timed out while step is verifying launched instance(s) are ready to be used. Please refer to Automation Service Troubleshooting Guide for more diagnosis details.`

Une étape de l'action `aws:runInstances` a expiré. Cela peut se produire si l'action de l'étape met plus de temps à s'exécuter que la valeur spécifiée pour `timeoutSeconds` dans l'étape. Pour résoudre ce problème, spécifiez une valeur plus longue pour `timeoutSeconds` . Si cela ne résout pas le problème, cherchez pourquoi l'étape est plus longue à s'exécuter que prévu.

AWS Systems Manager Modifier le calendrier

Systems Manager Modifier le calendrier vous permet de configurer des plages de dates et de temps pendant lesquelles les actions que vous spécifiez (par exemple, dans des documents Systems Manager Automation (p. 304)) peuvent ou non être effectuées dans votre compte AWS. Dans Modifier le calendrier, ces plages sont appelées des événements. Lorsque vous créez une Modifier le calendrier entrée, vous créez un Systems Manager document (p. 1211) du type `ChangeCalendar` . Dans Modifier le calendrier , le document stocke les données iCalendar 2.0 au format texte brut. Les événements que vous ajoutez à l'entrée Modifier le calendrier font partie du document.

Une entrée Modifier le calendrier peut être de l'un des deux types suivants :

`DEFAULT_OPEN` ou Ouvert par défaut

Lorsqu'une entrée de calendrier est ouverte par défaut, les actions peuvent s'exécuter par défaut, mais elles sont bloquées pendant les événements associés. Pendant les événements, l'état d'un calendrier `DEFAULT_OPEN` est `CLOSED`.

`DEFAULT_CLOSED` ou Fermé par défaut

Lorsqu'une entrée de calendrier est fermée par défaut, les actions qui suivent Modifier le calendrier ne s'exécutent pas par défaut, mais elles peuvent s'exécuter pendant les événements associés à l'entrée de calendrier. Pendant les événements, l'état d'un calendrier `DEFAULT_CLOSED` est `OPEN`.

À qui est destiné Modifier le calendrier ?

- Tout client AWS qui crée ou exécute des documents Systems Manager Automation.
- Administrateurs responsables de la cohérence, de la stabilité et du fonctionnement des configurations des instances gérées AWS Systems Manager.

Avantages d'Modifier le calendrier

Voici quelques avantages de Systems Manager Modifier le calendrier.

- Vérifier les modifications avant qu'elles soient appliquées

  Une entrée Modifier le calendrier peut vous aider à vous assurer que les modifications Automation potentiellement destructrices apportées à votre environnement sont passées en revue avant d'être appliquées.
- Appliquer les modifications uniquement pendant les heures appropriées

  Les entrées Modifier le calendrier permettent de préserver la stabilité de votre environnement pendant les heures des événements. Par exemple, vous pouvez créer une entrée Modifier le calendrier pour bloquer les modifications lorsque vous vous attendez à une forte demande sur vos ressources, par exemple, lors d'une conférence ou d'une promotion marketing publique. Une entrée de calendrier peut également bloquer les modifications lorsque vous vous attendez à un support administratif limité, par exemple, pendant les vacances ou les jours fériés. Vous pouvez utiliser une entrée de calendrier pour autoriser les modifications sauf pendant certaines périodes de la journée ou de la semaine où le support administratif est limité pour résoudre les échecs d'actions ou de déploiements.
- Obtenir l'état actuel ou à venir du calendrier

  Vous pouvez exécuter l'opération d'API `GetCalendarState` de Systems Manager pour afficher l'état actuel du calendrier, son état à une heure spécifique ou la prochaine fois qu'une modification de l'état du calendrier est planifiée.

Rubriques

- Configuration d'Modifier le calendrier (p. 681)
- Utilisation des Modifier le calendrier (p. 682)
- Ajouter des dépendances Modifier le calendrier aux documents Automation (p. 688)

Configuration d'Modifier le calendrier

Effectuez les opérations suivantes avant d'utiliser Modifier le calendrier.

Installer les derniers outils de ligne de commande

Installez les derniers outils de ligne de commande pour obtenir des informations d'état sur les calendriers.

Exigence	Description
AWS CLI	(Facultatif) Pour utiliser l'AWS CLI afin d'obtenir des informations d'état sur les calendriers, installez la version la plus récente de l'AWS CLI sur votre ordinateur local.

Exigence	Description
	Pour de plus amples informations sur l'installation ou la mise à niveau de l'interface de ligne de commande, veuillez consulter Installation de l'AWS Command Line Interface dans le AWS Command Line Interface Guide de l'utilisateur.
Outils AWS pour PowerShell	(Facultatif) Pour utiliser les Outils pour PowerShell afin d'obtenir des informations d'état sur les calendriers, installez la version la plus récente des Outils pour PowerShell sur votre ordinateur local. Pour plus d'informations sur l'installation ou la mise à niveau de l' Outils pour PowerShell , consultez Configuration des outils AWS pour Windows PowerShell ou des outils AWS pour PowerShell Core dans le Guide Outils AWS pour PowerShell de l'utilisateur .

Configuration d'autorisations

Pour créer, mettre à jour ou supprimer une entréeModifier le calendrier, y compris pour ajouter et supprimer des événements de l'entrée, une stratégie attachée à votre utilisateur ou rôle de service AWS Identity and Access Management (IAM) doit autoriser les actions suivantes.

- `ssm:CreateDocument`
- `ssm:DescribeDocument`
- `ssm:UpdateDocument`
- `ssm:DeleteDocument`

Pour obtenir des informations sur l'état actuel ou à venir du calendrier, une stratégie attachée à votre utilisateur ou rôle de service AWS Identity and Access Management (IAM) doit autoriser l'action suivante.

- `ssm:GetCalendarState`

Si votre compte utilisateur, votre groupe ou votre rôle IAM dispose des autorisations d'administrateur, vous avez accès à Modifier le calendrier. Si vous ne disposez pas des autorisations d'administrateur, un administrateur doit vous les donner en affectant la stratégie gérée `AmazonSSMFullAccess` ou une stratégie dotée d'autorisations de même type à votre compte, groupe ou rôle IAM.

Les entrées Modifier le calendrier appartenant à (c'est-à-dire créées par) d'autres comptes que le vôtre sont en lecture seule, même si elles sont partagées avec votre compte.

Utilisation des Modifier le calendrier

Vous pouvez utiliser la console AWS Systems Manager pour ajouter, gérer ou supprimer des entrées Modifier le calendrier. Utilisez l'API `GetCalendarState` ou la commande `get-calendar-state` de l'AWS CLI pour obtenir des informations sur l'état de Modifier le calendrier à un moment spécifique.

Rubriques

- Créer une entrée Modifier le calendrier (p. 683)
- Créer un événement Modifier le calendrier (p. 683)
- Mettre à jour un événement Modifier le calendrier (p. 684)

- Supprimer un événement Modifier le calendrier (p. 685)
- Mettre à jour une entrée Modifier le calendrier (p. 685)
- Partager une entrée Modifier le calendrier (p. 686)
- Supprimer une entrée Modifier le calendrier (p. 686)
- Obtenir l'état de Modifier le calendrier (p. 686)

Créer une entrée Modifier le calendrier

Lorsque vous créez une entrée AWS Systems ManagerModifier le calendrier, vous créez un document Systems Manager qui utilise le format `text`.

Créer une entrée Modifier le calendrier (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Modifier le calendrier.
3. Dans la page d'accueil Modifier le calendrier, choisissez Create calendar (Créer un calendrier).
4. Dans la page Create calendar (Créer un calendrier) dans Calendar details (Détails du calendrier), saisissez un nom pour votre entrée de calendrier. Les noms d'entrée de calendrier peuvent contenir des lettres, des chiffres, des points, des tirets et des traits de soulignement. Le nom doit être suffisamment précis pour permettre d'identifier immédiatement le but de l'entrée de calendrier. Voici un exemple **`support-off-hours`** : . Vous ne pouvez pas mettre à jour ce nom après avoir créé l'entrée de calendrier.
5. Dans Description, saisissez une description de votre entrée de calendrier.
6. Dans Calendar type (Type de calendrier), choisissez l'une des options suivantes :
   - Open by default (Ouvert par défaut) : le calendrier est ouvert (les actions Automation peuvent s'exécuter jusqu'au démarrage d'un événement), puis fermé pendant la durée d'un événement associé.
   - Closed by default (Fermé par défaut) : le calendrier est fermé (les actions Automation ne peuvent pas s'exécuter avant le démarrage d'un événement), mais ouvert pendant la durée d'un événement associé.
7. Choisissez Create calendar (Créer un calendrier).

   Une fois l'entrée de calendrier créée, Systems Manager affiche votre entrée de calendrier dans la liste Modifier le calendrier. Les colonnes affichent la version du calendrier et le numéro de compte AWS du propriétaire du calendrier. Votre entrée de calendrier ne peut ni empêcher ni autoriser des actions tant que vous n'avez pas ajouté au moins un événement. Pour plus d'informations sur l'ajout d'un événement, consultez Créer un événement Modifier le calendrier (p. 683).

Créer un événement Modifier le calendrier

Lorsque vous ajoutez un événement à une entrée Modifier le calendrier, vous spécifiez une période pendant laquelle l'action par défaut de l'entrée de calendrier est suspendue. Par exemple, si l'entrée de calendrier est de type Fermé par défaut, le calendrier est ouvert aux modifications pendant les événements. Dans cette version, vous ne pouvez créer un événement Modifier le calendrier qu'à l'aide de la console. Les événements sont ajoutés au document Systems Manager Modifier le calendrier que vous créez lorsque vous créez une entrée Modifier le calendrier.

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Modifier le calendrier.

3. Dans la liste Modifier le calendrier, choisissez le nom de l'entrée de calendrier à laquelle vous souhaitez ajouter un événement.
4. Dans la page des détails de l'entrée de calendrier, choisissez Create event (Créer un événement).
5. Dans la page Create scheduled event (Créer un événement planifié) dans Event details (Détails de l'événement), saisissez le nom complet de votre événement. Les noms d'événement peuvent contenir des lettres, des chiffres, des points, des tirets et des traits de soulignement. Le nom doit être suffisamment précis pour permettre d'identifier le but de l'événement. Voici un exemple **`nighttime-hours`** : . Vous ne pouvez pas mettre à jour ce nom après avoir créé l'événement.
6. Dans Description, saisissez une description de votre événement. Par exemple, **`The support team is not available during these hours`**.
7. Dans Event start date (Date de début de l'événement), saisissez ou choisissez un jour de démarrage de l'événement, au format `MM/DD/YYYY`, puis saisissez une heure de démarrage de l'événement le jour spécifié, au format `hh:mm:ss` (heures, minutes et secondes).
8. Dans Event end date (Date de fin de l'événement), saisissez ou choisissez un jour de fin de l'événement, au format `MM/DD/YYYY`, puis saisissez une heure de fin de l'événement le jour spécifié, au format `hh:mm:ss` (heures, minutes et secondes).
9. Dans Schedule time zone (Fuseau horaire de planification), choisissez un fuseau horaire qui s'applique aux heures de début et de fin de l'événement. Vous pouvez saisir une partie d'un nom de ville ou d'une différence de fuseau horaire par rapport à l'heure GMT (heure moyenne de Greenwich) pour trouver un fuseau horaire plus rapidement. La valeur par défaut est UTC (temps universel coordonné).
10. Pour créer un événement qui se répète chaque jour, chaque semaine ou chaque mois, activez l'option Recurrence (Récurrence).
11. Choisissez Create scheduled event (Créer un événement planifié). Le nouvel événement est ajouté à votre entrée de calendrier et s'affiche sous l'onglet Events (Événements) de la page des détails de l'entrée de calendrier.

Mettre à jour un événement Modifier le calendrier

Mettre à jour un événement Modifier le calendrier (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Modifier le calendrier.
3. Dans la liste Modifier le calendrier, choisissez le nom de l'entrée de calendrier pour laquelle vous souhaitez modifier un événement.
4. Dans la page des détails de l'entrée de calendrier, choisissez Events (Événements).
5. Dans la page du calendrier, choisissez l'événement que vous souhaitez modifier. Utilisez les boutons situés en haut à gauche pour reculer ou avancer d'un an, ou reculer ou avancer d'un mois. Modifiez le fuseau horaire, si nécessaire, en choisissant le fuseau horaire approprié dans la liste déroulante.
6. Dans Event details (Détails de l'événement), vous pouvez modifier le texte du champ Description. Vous ne pouvez pas modifier le nom de l'événement.
7. Pour modifier la valeur du champ Event start date (Date de début de l'événement), saisissez ou choisissez un jour de démarrage de l'événement, au format `MM/DD/YYYY`, puis saisissez une heure de démarrage de l'événement le jour spécifié, au format `hh:mm:ss` (heures, minutes et secondes).
8. Pour modifier la valeur du champ Event end date (Date de fin de l'événement), saisissez ou choisissez un jour de fin de l'événement, au format `MM/DD/YYYY`, puis saisissez une heure de fin de l'événement le jour spécifié, au format `hh:mm:ss` (heures, minutes et secondes).
9. Pour modifier la valeur du champ Schedule time zone (Fuseau horaire de planification), choisissez un fuseau horaire qui s'applique aux heures de début et de fin de l'événement. Vous pouvez saisir une partie d'un nom de ville ou d'une différence de fuseau horaire par rapport à l'heure GMT (heure moyenne de Greenwich) pour trouver un fuseau horaire plus rapidement. La valeur par défaut est UTC (temps universel coordonné).

10. Pour mettre à jour un événement afin qu'il se répète chaque jour, chaque semaine ou chaque mois, activez l'option Recurrence (Récurrence).
11. Choisissez Update scheduled event (Mettre à jour l'événement planifié). Vos modifications s'affichent sous l'onglet Events (Événements) de la page des détails de l'entrée de calendrier. Choisissez l'événement que vous avez mis à jour pour afficher vos modifications.

Supprimer un événement Modifier le calendrier

Vous pouvez supprimer un événement à l'aide d'AWS Management Console.

Supprimer un événement Modifier le calendrier (console)

Vous pouvez supprimer un événement à la fois à l'aide d'AWS Management Console.

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Modifier le calendrier.
3. Dans la liste Modifier le calendrier, choisissez le nom de l'entrée de calendrier à partir de laquelle vous souhaitez supprimer un événement.
4. Dans la page des détails de l'entrée de calendrier, choisissez Events (Événements).
5. Dans la page du calendrier, choisissez l'événement que vous souhaitez supprimer. Utilisez les boutons situés en haut à gauche pour faire reculer ou avancer le calendrier d'un an, ou pour le faire reculer ou avancer d'un mois. Modifiez le fuseau horaire, si nécessaire, en choisissant le fuseau horaire approprié dans la liste déroulante.
6. Dans la page Event details (Détails de l'événement), choisissez Delete (Supprimer). Lorsque vous êtes invité à confirmer que vous souhaitez supprimer l'événement, choisissez Delete (Supprimer).

Mettre à jour une entrée Modifier le calendrier

Mettre à jour une entrée Modifier le calendrier (console)

Vous pouvez mettre à jour la description d'une entrée Modifier le calendrier, mais pas son nom. Bien que vous puissiez modifier l'état par défaut d'une entrée de calendrier, sachez que cela inverse le comportement des actions de modification lors des événements qui sont associés à cette entrée de calendrier. Par exemple, si vous remplacez l'état Ouvert par défaut d'un calendrier par Fermé par défaut, des modifications indésirables peuvent être effectuées pendant les périodes d'événement lorsque les utilisateurs qui ont créé les événements associés n'attendent pas de modifications.

Lorsque vous mettez à jour une entrée Modifier le calendrier, vous modifiez le document Systems Manager Modifier le calendrier que vous avez créé lors de la création de cette entrée Modifier le calendrier.

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Modifier le calendrier.
3. Dans la liste Modifier le calendrier, choisissez le nom de l'entrée de calendrier à mettre à jour.
4. Dans la page des détails de l'entrée de calendrier, choisissez Edit (Modifier).
5. Dans Description, vous pouvez modifier le texte de la description. Vous ne pouvez pas modifier le nom d'une entrée Modifier le calendrier.
6. Pour modifier l'état du calendrier, choisissez une autre valeur dans Calendar type (Type de calendrier). Sachez que cela inverse le comportement des actions de modification lors des événements qui sont associés à l'entrée de calendrier. Avant de modifier le type de calendrier, vous devez vérifier auprès des autres utilisateurs Modifier le calendrier que ce changement de type de calendrier n'autorise pas des modifications indésirables lors des événements qu'ils ont créés.

- Open by default (Ouvert par défaut) : le calendrier est ouvert (les actions Automation peuvent s'exécuter jusqu'au démarrage d'un événement), puis fermé pendant la durée d'un événement associé.
- Closed by default (Fermé par défaut) : le calendrier est fermé (les actions Automation ne peuvent pas s'exécuter avant le démarrage d'un événement), mais ouvert pendant la durée d'un événement associé.

7. Choisissez Save.

   Votre entrée de calendrier ne peut ni empêcher ni autoriser des actions tant que vous n'avez pas ajouté au moins un événement. Pour plus d'informations sur l'ajout d'un événement, consultez Créer un événement Modifier le calendrier (p. 683).

Partager une entrée Modifier le calendrier

Vous pouvez partager une entrée Modifier le calendrier avec d'autres comptes AWS à l'aide d'AWS Management Console. Lorsque vous partagez un calendrier, ce calendrier est en lecture seule pour les utilisateurs du compte partagé.

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Modifier le calendrier.
3. Dans la liste Modifier le calendrier, choisissez le nom de l'entrée de calendrier que vous souhaitez partager.
4. Dans la page des détails de l'entrée de calendrier, choisissez Share (Partager).
5. Dans Share calendar (Partager le calendrier), pour Account ID (ID de compte), saisissez le numéro d'identification d'un compte AWS valide, puis choisissez Share (Partager).

   Les utilisateurs du compte partagé peuvent lire l'entrée Modifier le calendrier, mais ne peuvent y apporter aucune modification.

Supprimer une entrée Modifier le calendrier

Vous pouvez supprimer une entrée Modifier le calendrier à l'aide d'AWS Management Console ou de l'AWS CLI. La suppression d'une entrée Modifier le calendrier supprime tous les événements associés.

Supprimer une entrée Modifier le calendrier (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Modifier le calendrier.
3. Dans la liste Modifier le calendrier, choisissez le nom de l'entrée de calendrier à supprimer.
4. Dans la page des détails de l'entrée de calendrier, choisissez Delete (Supprimer). Lorsque vous êtes invité à confirmer que vous souhaitez supprimer l'entrée de calendrier, choisissez Delete (Supprimer).

Obtenir l'état de Modifier le calendrier

Vous pouvez obtenir l'état global du calendrier ou son état à un moment spécifique. Vous pouvez également afficher la prochaine fois que l'état du calendrier passera de `OPEN` à `CLOSED`, ou inversement.

Vous ne pouvez effectuer cette tâche qu'à l'aide de l'API `GetCalendarState`. La procédure décrite dans cette section utilise l'AWS CLI.

- Exécutez la commande suivante pour afficher l'état d'une ou plusieurs entrées de calendrier à un moment spécifique. Le paramètre `--calendar-names` est obligatoire, mais `--at-time` est facultatif.

  Linux

  ```
  aws ssm get-calendar-state \
      --calendar-names "Calendar_name_or_document_ARN_1"
   "Calendar_name_or_document_ARN_2" \
      --at-time "ISO_8601_time_format"
  ```

  Voici un exemple pour .

  ```
  aws ssm get-calendar-state \
      --calendar-names "arn:aws:ssm:us-east-2:123456789012:document/
  MyChangeCalendarDocument" "arn:aws:ssm:us-east-2:123456789012:document/
  SupportOffHours" \
      --at-time "2020-07-30T11:05:14-0700"
  ```

  Windows

  ```
  aws ssm get-calendar-state ^
      --calendar-names "Calendar_name_or_document_ARN_1"
   "Calendar_name_or_document_ARN_2" ^
      --at-time "ISO_8601_time_format"
  ```

  Voici un exemple pour .

  ```
  aws ssm get-calendar-state ^
      --calendar-names "arn:aws:ssm:us-east-2:123456789012:document/
  MyChangeCalendarDocument" "arn:aws:ssm:us-east-2:123456789012:document/
  SupportOffHours" ^
      --at-time "2020-07-30T11:05:14-0700"
  ```

  La commande renvoie des informations telles que les suivantes.

  ```
  {
      "State": "OPEN",
      "AtTime": "2020-07-30T16:18:18Z",
      "NextTransitionTime": "2020-07-31T00:00:00Z"
  }
  ```

  Les résultats indiquent l'état du calendrier (s'il est de type `DEFAULT_OPEN` ou `DEFAULT_CLOSED`) pour les entrées de calendrier spécifiées appartenant à votre compte ou partagées avec celui-ci, au moment spécifié par la valeur de `--at-time`, et l'heure de la transition suivante. Si vous n'ajoutez pas le paramètre `--at-time`, l'heure actuelle est utilisée.

  > **Note**
  >
  > Si vous spécifiez plusieurs calendriers dans une demande, la commande renvoie le statut uniquement si tous les calendriers de la demande sont ouverts.`OPEN` Si un ou plusieurs calendriers de la demande sont fermés, le statut renvoyé est `CLOSED` .

Ajouter des dépendances Modifier le calendrier aux documents Automation

Pour que les actions Automation respectent Systems Manager Modifier le calendrier, ajoutez dans un document Automation une étape qui utilise l'action `aws:assertAwsResourceProperty` (p. 395). Configurez l'action pour exécuter `GetCalendarState` afin de vérifier qu'une entrée de calendrier spécifique est à l'état souhaité (`OPEN` ou `CLOSED`). Le document Automation n'est autorisé à passer à l'étape suivante que si l'état du calendrier est `OPEN` . Voici un exemple d'extrait YAML d'un document Automation qui ne peut pas passer à l'étape suivante, `LaunchInstance` , sauf si l'état du calendrier correspond à `OPEN` , l'état spécifié dans `DesiredValues` .

```
mainSteps:
  - name: MyCheckCalendarStateStep
    action: 'aws:assertAwsResourceProperty'
    inputs:
      Service: ssm
      Api: GetCalendarState
      CalendarNames: 'arn:aws:ssm:us-east-1:123456789012:document/SaleDays'
      PropertySelector: '$.State'
      DesiredValues:
      - OPEN
    description: "Use GetCalendarState to determine whether a calendar is open or closed."
    nextStep: LaunchInstance
  - name: LaunchInstance
    action: 'aws:executeScript'
    inputs:
      Runtime: python3.6
...
```

AWS Systems Manager Fenêtres de maintenance

> **Note**
>
> State Manager et Fenêtres de maintenance peuvent effectuer des types similaires de mise à jour sur vos instances gérées. Votre choix dépend de la nécessité d'automatiser la conformité du système ou d'effectuer des tâches prioritaires et urgentes uniquement pendant les périodes que vous spécifiez.
> Pour plus d'informations, consultez Choisir entre State Manager etFenêtres de maintenance (p. 1405).

Les Fenêtres de maintenance AWS Systems Manager vous permettent de définir un planning d'exécution d'actions potentiellement perturbatrices sur vos instances, telles que l'application de correctifs à un système d'exploitation, la mise à jour de pilotes ou l'installation de logiciels ou de correctifs. Les Fenêtres de maintenance vous permettent également de planifier des actions sur de nombreux autres types de ressources AWS, comme des compartiments Amazon Simple Storage Service (Amazon S3), des files d'attente Amazon Simple Queue Service (Amazon SQS), des clés AWS Key Management Service (AWS KMS), et bien plus encore. Pour une liste complète des types de ressources pris en charge que vous pouvez inclure dans une fenêtre de maintenance cible, consultez Ressources prises en charge pour Groupes de ressources AWS dans le Guide de l'utilisateur Groupes de ressources AWS.

Chaque fenêtre de maintenance dispose d'une planification, d'une durée maximale, d'un ensemble de cibles enregistrées (instances et autres ressources AWS traitées) et d'un ensemble de tâches enregistrées. Vous pouvez ajouter des balises à vos fenêtres de maintenance lorsque vous les créez ou les mettez à jour. (Les balises sont des clés qui facilitent l'identification et le tri de vos ressources au sein de votre

organisation.) Vous pouvez également spécifier les dates avant ou après lesquelles une fenêtre de maintenance ne doit pas s'exécuter, et vous pouvez spécifier le fuseau horaire international sur lequel baser la planification de la fenêtre de maintenance.

Pour obtenir une explication des relations entre les différentes options liées à la planification pour les fenêtres de maintenance, consultez Référence : Options de planification de la fenêtre de maintenance et de période active (p. 772).

Pour de plus amples informations sur l'utilisation de l'option `--schedule`, veuillez consulter Référence : Expressions cron et rate pour Systems Manager (p. 1392).

Types de tâches pris en charge

Les fenêtres de maintenance prennent en charge l'exécution de quatre types de tâches :

- Commandes de Fonctionnalité Exécuter la commande Systems Manager

  Pour plus d'informations sur Fonctionnalité Exécuter la commande, consultez AWS Systems Manager Fonctionnalité Exécuter la commande (p. 941).
- Flux de travail d'automatisation Systems Manager

  Pour de plus amples informations sur les flux de travail Automation, veuillez consulter Automatisation AWS Systems Manager (p. 304).
- Fonctions AWS Lambda

  Pour de plus amples informations sur les fonctions Lambda, veuillez consulter Utilisation des fonctions Lambda dans le AWS Lambda Developer Guide.
- Tâches AWS Step Functions

  Pour de plus amples informations sur Step Functions, veuillez consulter Manuel du développeur AWS Step Functions.

Cela signifie que vous pouvez utiliser les fenêtres de maintenance pour effectuer les tâches suivantes sur vos cibles sélectionnées :

- Installer ou mettre à jour des applications.
- Appliquer des correctifs.
- Installer ou mettre à jour l'Agent SSM.
- Exécutez PowerShell des commandes et des scripts shell Linux à l'aide d'une Systems Manager Fonctionnalité Exécuter la commande tâche.
- Créer des Amazon Machine Images (AMI), amorcer des logiciels et configurer des instances à l'aide d'une tâche Systems Manager Automation.
- Exécuter des fonctions AWS Lambda déclenchant des actions supplémentaires, telles que l'analyse de vos instances pour rechercher les mises à jour de correctif.
- Exécuter des machines d'état AWS Step Functions pour effectuer des tâches telles que la suppression d'une instance d'un environnement Elastic Load Balancing, l'application d'un correctif à l'instance, puis le rajout de l'instance à l'environnement Elastic Load Balancing.
- Cibler les instances qui sont hors ligne en spécifiant un groupe de ressources AWS en tant que cible.

Prise en charge de Amazon EventBridge

Cette fonction Systems Manager est prise en charge en tant que type d'événement dans les règles EventBridge. Pour plus d'informations, consultez Surveillance des événements Systems Manager avec Amazon EventBridge (p. 1340) et Référence : modèles et types Amazon EventBridge d'événements pourSystems Manager (p. 1387).

Table des matières

- Configuration d'Fenêtres de maintenance (p. 690)
- Utilisation des fenêtres de maintenance (console) (p. 707)
- Didacticiels sur les fenêtres de maintenance Systems Manager (AWS CLI) (p. 715)
- Procédures pas à pas d'une fenêtre de maintenance (p. 763)
- Référence : Options de planification de la fenêtre de maintenance et de période active (p. 772)
- Fenêtres de maintenance de dépannage (p. 776)

Configuration d'Fenêtres de maintenance

Pour que les utilisateurs de votre compte puissent créer et planifier des tâches de fenêtre de maintenance, ils doivent bénéficier des autorisations nécessaires. Pour accorder ces autorisations aux utilisateurs, un administrateur doit exécuter les deux tâches suivantes :

Tâche 1 : Configurer les autorisations d'instance

Fournissez au service Fenêtres de maintenance les autorisations AWS Identity and Access Management (IAM) nécessaires pour exécuter des tâches de fenêtre de maintenance sur vos instances en effectuant l'une des actions suivantes :

- Créer un rôle de service personnalisé pour les tâches de fenêtre de maintenance
- Créer un rôle lié à un service pour AWS Systems Manager

Vous spécifiez l'un de ces rôles dans le cadre de la configuration, lorsque vous créez une tâche de fenêtre de maintenance. Cela permet à Systems Manager d'exécuter des tâches dans les fenêtres de maintenance en votre nom.

> **Note**
>
> Un rôle lié à un service pour Systems Manager peut déjà avoir été créé dans votre compte. Actuellement, le rôle lié à un service fournit également des autorisations pour la fonction d'inventaire.

Pour choisir plus facilement d'utiliser un rôle de service personnalisé ou le rôle lié à un service Systems Manager avec une tâche de fenêtre de maintenance, consultez Dois-je utiliser un rôle lié à un service ou un rôle de service personnalisé pour exécuter des tâches de fenêtre de maintenance ? (p. 691).

Tâche 2 : Configurer des autorisations pour les utilisateurs autorisés à enregistrer des tâches de fenêtre de maintenance

Accordez `iam:PassRole` des autorisations aux utilisateurs de votre compte qui attribuent des tâches aux fenêtres de maintenance. Cela leur permet de transmettre le rôle au service de fenêtre de maintenance. Sans cette autorisation explicite, un utilisateur ne peut pas attribuer de tâches à une fenêtre de maintenance lorsqu'il utilise un rôle de service personnalisé pour exécuter des tâches de fenêtre de maintenance.

Tâche 3 : Configurer des autorisations pour les utilisateurs qui ne sont pas autorisés à enregistrer des tâches de fenêtre de maintenance

Refusez `ssm:RegisterTaskWithMaintenanceWindow` les autorisations pour les utilisateurs de votre compte pour lesquels vous ne souhaitez pas enregistrer des tâches avec des fenêtres de maintenance. Cela empêche les utilisateurs d'enregistrer une tâche de fenêtre de maintenance en utilisant le rôle lié à un service dans une demande d'enregistrement de tâche de fenêtre de maintenance.

Avant de commencer

Pour effectuer les tâches de la section, vous avez besoin de l'une des ressources suivantes ou des deux.

- Vous attribuez des autorisations à des utilisateurs ou groupes IAM. Ces utilisateurs ou groupes doivent avoir déjà obtenu des autorisations générales pour utiliser les fenêtres de maintenance. Cela peut être fait en affectant la stratégie IAM `AmazonSSMFullAccess` aux utilisateurs ou groupes, ou en créant et en affectant une stratégie IAM qui fournit un ensemble plus petit d'autorisations d'accès pour Systems Manager qui couvre les tâches de la fenêtre de maintenance. Pour de plus amples informations, veuillez consulter Créer des groupes d'utilisateurs (p. 31) et Créer des utilisateurs et attribuer des autorisations (p. 32).
- (Facultatif) Pour les fenêtres de maintenance qui exécutent des tâches Fonctionnalité Exécuter la commande, vous pouvez choisir d'envoyer des notifications d'état Amazon Simple Notification Service (Amazon SNS). Pour plus d'informations sur la configuration des notifications Amazon SNS pour Systems Manager, notamment sur la création d'un rôle IAM à utiliser pour l'envoi de notifications SNS, reportez-vous à la section Surveillance des changements de statut Systems Manager à l'aide de notifications Amazon SNS (p. 1344).

Dois-je utiliser un rôle lié à un service ou un rôle de service personnalisé pour exécuter des tâches de fenêtre de maintenance ?

Pour exécuter des tâches de maintenance sur vos instances cibles, le service Fenêtres de maintenance doit être autorisé à accéder aux tâches sur vos instances et à les exécuter. Vous pouvez fournir cette autorisation en spécifiant un rôle de service personnalisé ou le rôle lié à un service Systems Manager dans le cadre de la configuration d'une tâche.

Le type de rôle que vous devez choisir dépend des facteurs suivants :

Rôle de service personnalisé : Utilisez un rôle de service personnalisé pour les tâches de fenêtre de maintenance dans les cas suivants :

- Si vous souhaitez utiliser un ensemble d'autorisations plus restrictif que celles fournies par le rôle lié à un service. Le rôle lié à un service prend en charge des contraintes très limitées au niveau des ressources. Par exemple, supposons que vous voulez autoriser les tâches de fenêtre de maintenance à s'exécuter sur un ensemble limité d'instances, ou que vous souhaitez autoriser uniquement certains documents SSM à s'exécuter sur vos instances cibles. Dans ces cas, vous spécifiez des autorisations plus strictes dans un rôle de service personnalisé.
- Si vous souhaitez utiliser un ensemble d'autorisations plus permissif ou étendu que celles fournies par le rôle lié à un service. Certaines actions dans les documents d'automatisation nécessitent des autorisations étendues.

  Par exemple, certaines actions d'automatisation utilisent des piles AWS CloudFormation. Par conséquent, les autorisations `cloudformation:CreateStack`, `cloudformation:DescribeStack` et `cloudformation:DeleteStack` sont requises.

  Autre exemple : le document Automation `AWS-CopySnapshot` nécessite l'autorisation de créer un instantané Amazon Elastic Block Store (Amazon EBS) et, par conséquent, le rôle de service a besoin de l'autorisation `ec2:CreateSnapshot` . Cette autorisation n'est pas incluse dans le rôle lié à un service pour Systems Manager .

  Pour plus d'informations sur le rôle des autorisations requises par les documents d'automatisation, consultez les descriptions du document dans Référence sur les détails du document Automation (p. 511).

Rôle lié à un Systems Manager service : Nous vous recommandons d'utiliser un rôle lié à un Systems Manager service dans tous les autres cas.

Pour de plus amples informations sur le rôle lié à un service Systems Manager, veuillez consulter Utilisation des rôles liés à un service pour Systems Manager (p. 1312).

Rubriques

- Contrôle de l'accès aux fenêtres de maintenance (console) (p. 692)
- Contrôler l'accès aux fenêtres de maintenance (AWS CLI) (p. 697)
- Contrôler l'accès aux fenêtres de maintenance (Outils pour Windows PowerShell) (p. 703)

Contrôle de l'accès aux fenêtres de maintenance (console)

Les procédures suivantes expliquent comment utiliser la console AWS Systems Manager pour créer les rôles et autorisations requis pour des fenêtres de maintenance.

Rubriques

- Tâche 1 : (Facultatif) Créer un rôle de service personnalisé pour les fenêtres de maintenance (console) (p. 692)
- Tâche 2 : Configurer des autorisations pour les utilisateurs autorisés à enregistrer des tâches de fenêtre de maintenance (console) (p. 694)
- Tâche 3 : Configurer des autorisations pour les utilisateurs qui ne sont pas autorisés à enregistrer des tâches de fenêtre de maintenance (console) (p. 696)

Tâche 1 : (Facultatif) Créer un rôle de service personnalisé pour les fenêtres de maintenance (console)

Utilisez la procédure suivante pour créer un rôle de service personnalisé pour la fonctionnalité Fenêtres de maintenance afin que Systems Manager puisse exécuter des tâches en votre nom.

> **Important**
>
> Un rôle de service personnalisé n'est pas requis si vous choisissez d'utiliser un rôle lié à un service Systems Manager pour laisser les fenêtres de maintenance exécuter des tâches en votre nom. Si vous n'avez pas de rôle lié à un service Systems Manager dans votre compte, vous pouvez le créer lorsque vous créez ou mettez à jour une tâche de fenêtre de maintenance à l'aide de la console Systems Manager. Pour de plus amples informations, consultez les rubriques suivantes :
>
> - Dois-je utiliser un rôle lié à un service ou un rôle de service personnalisé pour exécuter des tâches de fenêtre de maintenance ? (p. 691)
> - Utilisation des rôles liés à un service pour Systems Manager (p. 1312)
> - Attribuer des tâches à une fenêtre de maintenance (console) (p. 712)

Pour créer un rôle de service personnalisé (console)

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le volet de navigation, choisissez Rôles, puis Créer un rôle.
3. Effectuez les sélections suivantes :
   1. Sélectionnez le type de zone de l'entité de confiance : Service AWS
   2. Choisissez le service qui utilisera ce rôle : zone Systems Manager
4. Choisissez Suivant : Autorisations.
5. Dans la liste des stratégies, cochez la case en regard de AmazonSSMMaintenanceWindowRole , puis choisissez Suivant : Balises.

6. (Facultatif) Ajoutez une ou plusieurs paires clé-valeur de balise pour organiser, suivre ou contrôler l'accès pour ce rôle, puis choisissez Suivant : Passez en revue.
7. Dans Nom du rôle, entrez un nom qui identifie ce rôle en tant que rôle Fenêtres de maintenance ; par exemple **`my-maintenance-window-role`**
8. (Facultatif) Modifiez la description du rôle par défaut pour refléter l'objectif de ce rôle. Exemple : **`Performs maintenance window tasks on your behalf`**.
9. Sélectionnez Créer un rôle. Le système vous renvoie à la page Rôles.
10. Choisissez le nom du rôle que vous venez de créer.
11. Choisissez l'onglet Relations d'approbation, puis Modifier la relation d'approbation.
12. Vérifiez que la stratégie suivante apparaît dans le champ Document de stratégie.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": "ssm.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

13. Choisissez Mettre à jour la stratégie de confiance, puis copiez ou notez le nom du rôle et la valeur du champ ARN de rôle sur la page Récapitulatif. Vous spécifiez cette information lorsque vous créez votre fenêtre de maintenance.
14. (Facultatif) Si vous prévoyez de configurer une fenêtre de maintenance pour envoyer des notifications sur les statuts de commande à l'aide d' Amazon Simple Notification Service (Amazon SNS), lors de l'exécution via une tâche de Fonctionnalité Exécuter la commande commande, procédez comme suit :

    1. Choisissez l'onglet Autorisations.
    2. Choisissez Add inline policy (Ajouter une stratégie en ligne), puis choisissez l'onglet JSON.
    3. Collez ce qui suit dans Document de stratégie.

    ```
    {
      "Version": "2012-10-17",
      "Statement": [
        {
          "Effect": "Allow",
          "Action": "iam:PassRole",
          "Resource": "sns-access-role-arn"
        }
      ]
    }
    ```

    Remplacez *`sns-access-role-arn`* par l'ARN du IAM rôle existant à utiliser pour envoyer des notifications Amazon Simple Notification Service (Amazon SNS) liées à la fenêtre de maintenance, au format par exemple :`arn:aws:iam::`*`account-id`*`:role/`*`role-name`*`.` `arn:aws:iam::123456789012:role/my-sns-access-role`. Pour plus d'informations sur la configuration Amazon SNS des notifications pour Systems Manager , notamment sur la création d'un IAM rôle à utiliser pour l'envoi de notifications SNS, consultez Surveillance des changements de statut Systems Manager à l'aide de notifications Amazon SNS (p. 1344) .

Note

Dans la console Systems Manager, cet ARN est sélectionné dans la liste IAM Role (Rôle) sur la page Register run command task (Enregistrer une tâche d'exécution de commande). Pour plus d'informations, consultez la section Attribuer des tâches à une fenêtre de maintenance (console) (p. 712). Dans l'API Systems Manager, cet ARN est saisi en tant que valeur de ServiceRoleArn dans la demande SendCommand.

4. Choisissez Examiner une stratégie.
5. Dans la zone Nom, entrez un nom pour l'identifier en tant que stratégie afin d'autoriser l'envoi de notifications Amazon SNS.

15. Choisissez Create policy (Créer une stratégie).

Tâche 2 : Configurer des autorisations pour les utilisateurs autorisés à enregistrer des tâches de fenêtre de maintenance (console)

Lorsque vous enregistrez une tâche avec une fenêtre de maintenance, vous spécifiez un rôle de service personnalisé ou un rôle lié à un service Systems Manager pour exécuter les opérations de tâche réelles. Il s'agit du rôle assumé par le service lorsqu'il exécutera des tâches en votre nom. Avant cela, pour enregistrer la tâche elle-même, vous devez affecter la stratégie IAM PassRole à un compte utilisateur IAM ou à un groupe IAM. Cela permet à l'utilisateur IAM ou au groupe IAM de spécifier, dans le cadre de l'enregistrement de ces tâches dans la fenêtre de maintenance, le rôle à utiliser lors de l'exécution des tâches. Pour de plus amples informations, veuillez consulter Octroi d'autorisations à un utilisateur pour transférer un rôle à un service AWS dans le IAM Guide de l'utilisateur.

Selon que vous attribuez l'autorisation `iam:Passrole` à un utilisateur individuel ou à un groupe, utilisez l'une des procédures suivantes pour fournir les autorisations minimales requises pour enregistrer des tâches avec une fenêtre de maintenance.

Pour configurer des autorisations pour les utilisateurs autorisés à enregistrer des tâches de fenêtre de maintenance (console)

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Choisissez Utilisateurs, puis le nom du compte utilisateur à mettre à jour.
3. Sous les onglets Autorisations, dans la liste de stratégies, vérifiez si la stratégie `AmazonSSMFullAccess` est répertoriée ou s'il existe une stratégie comparable qui accorde à l'utilisateur IAM l'autorisation d'appeler l'API Systems Manager. Ajoutez cette autorisation si elle n'est pas déjà incluse. Pour de plus amples informations, veuillez consulter Ajout et suppression de stratégies IAM (console) dans le IAM Guide de l'utilisateur.
4. Choisissez Add inline policy (Ajouter une stratégie en ligne), puis choisissez l'onglet JSON.
5. Collez ce qui suit dans Document de stratégie.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "custom-role-arn"
        },
        {
            "Effect": "Allow",
            "Action": "iam:ListRoles",
            "Resource": "arn:aws:iam::account-id:role/"
        },
        {
            "Effect": "Allow",
```

```
            "Action": "iam:ListRoles",
            "Resource": "arn:aws:iam::account-id:role/aws-service-role/
ssm.amazonaws.com/"
        }
    ]
}
```

Remplacez *custom-role-arn* avec l'ARN du rôle de fenêtre de maintenance personnalisé que vous avez créé précédemment, par exemple `arn:aws:iam::123456789012:role/my-maintenance-window-role` .

Remplacez *account-id* dans les deux `iam:ListRoles` autorisations avec l'ID de votre compte AWS. L'ajout de cette autorisation pour la ressource `arn:aws:iam::account-id:role/` permet à un utilisateur d'afficher et de choisir parmi les rôles client dans la console lorsqu'il crée une tâche de fenêtre de maintenance. L'ajout de cette autorisation pour `arn:aws:iam::account-id:role/aws-service-role/ssm.amazonaws.com/` permet à un utilisateur de choisir le rôle lié au service Systems Manager dans la console lorsqu'il crée une tâche de fenêtre de maintenance.

6. Choisissez Examiner une stratégie.
7. Sur la page Examiner une stratégie, entrez un nom dans la zone Nom pour identifier cette PassRole stratégie, par exemple **`my-iam-passrole-policy`** , puis choisissez Créer une stratégie.

Pour configurer des autorisations pour les groupes autorisés à enregistrer des tâches de fenêtre de maintenance (console)

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation, choisissez Groupes .
3. Dans la liste de groupes, sélectionnez le nom du groupe auquel accorder l'autorisation `iam:PassRole`.
4. Sous l'onglet Autorisations dans la section Stratégies intégrées, effectuez l'une des opérations suivantes :
   - Si aucune stratégie en ligne n'a été encore ajoutée, choisissez cliquez ici.
   - Si une ou plusieurs stratégies en ligne ont été ajoutés, choisissez Créer une stratégie de groupe.
5. Choisissez Stratégie personnalisée, puis Sélectionner.
6. Pour Policy Name (Nom de la stratégie), entrez un nom pour l'identifier en tant que stratégie de fenêtres de maintenance pour votre groupe, par exemple PassRole .**`my-group-iam-passrole-policy`**
7. Collez ce qui suit dans Document de stratégie.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "custom-role-arn"
        },
        {
            "Effect": "Allow",
            "Action": "iam:ListRoles",
            "Resource": "arn:aws:iam::account-id:role/"
        },
        {
            "Effect": "Allow",
            "Action": "iam:ListRoles",
            "Resource": "arn:aws:iam::account-id:role/aws-service-role/
ssm.amazonaws.com/"
```

```
        }
    ]
}
```

Remplacez *custom-role-arn* avec l'ARN du rôle de fenêtre de maintenance personnalisé que vous avez créé précédemment, par exemple `arn:aws:iam::123456789012:role/my-maintenance-window-role` .

Remplacez *account-id* dans les deux `iam:ListRoles` autorisations avec l'ID de votre compte AWS. L'ajout de cette autorisation pour la ressource `arn:aws:iam::`*account-id*`:role/` permet aux utilisateurs du groupe d'afficher et de choisir parmi les rôles clients dans la console lorsqu'ils créent une tâche de fenêtre de maintenance. L'ajout de cette autorisation pour `arn:aws:iam::`*account-id*`:role/aws-service-role/ssm.amazonaws.com/` permet aux utilisateurs du groupe de choisir le rôle lié au service Systems Manager dans la console lorsqu'ils créent une tâche de fenêtre de maintenance.

8. Choisissez Apply Policy (Appliquer la stratégie).

Tâche 3 : Configurer des autorisations pour les utilisateurs qui ne sont pas autorisés à enregistrer des tâches de fenêtre de maintenance (console)

Selon que vous refusez l'`ssm:RegisterTaskWithMaintenanceWindow`autorisation pour un utilisateur individuel ou un groupe, utilisez l'une des procédures suivantes pour empêcher les utilisateurs d'enregistrer des tâches avec une fenêtre de maintenance.

Pour configurer des autorisations pour les utilisateurs qui ne sont pas autorisés à enregistrer des tâches de fenêtre de maintenance (console)

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Choisissez Utilisateurs, puis le nom du compte utilisateur à mettre à jour.
3. Choisissez Add inline policy (Ajouter une stratégie en ligne), puis choisissez l'onglet JSON.
4. Collez ce qui suit dans Document de stratégie.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": "ssm:RegisterTaskWithMaintenanceWindow",
            "Resource": "*"
        }
    ]
}
```

5. Choisissez Examiner une stratégie.
6. Sur la page Vérifier la stratégie, entrez un nom dans la zone Nom pour identifier cette stratégie, par exemple **`my-deny-mw-tasks-policy`** , puis choisissez Créer une stratégie.

Pour configurer des autorisations pour les groupes qui ne sont pas autorisés à enregistrer des tâches de fenêtre de maintenance (console)

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation, choisissez Groupes .
3. Dans la liste des groupes, sélectionnez le nom du groupe dont vous souhaitez refuser l`ssm:RegisterTaskWithMaintenanceWindow`'autorisation.

4. Sous l'onglet Autorisations dans la section Stratégies intégrées, effectuez l'une des opérations suivantes :
   - Si aucune stratégie en ligne n'a été encore ajoutée, choisissez cliquez ici.
   - Si une ou plusieurs stratégies en ligne ont été ajoutés, choisissez Créer une stratégie de groupe.
5. Choisissez Stratégie personnalisée, puis Sélectionner.
6. Pour Policy Name (Nom de la stratégie), entrez un nom pour l'identifier en tant que stratégie de fenêtres de maintenance pour votre groupe, par exemple PassRole .**`my-group-deny-mw-tasks-policy`**
7. Collez ce qui suit dans Document de stratégie.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": "ssm:RegisterTaskWithMaintenanceWindow",
            "Resource": "*"
        }
    ]
}
```

8. Choisissez Apply Policy (Appliquer la stratégie).

Contrôler l'accès aux fenêtres de maintenance (AWS CLI)

Les procédures suivantes expliquent comment utiliser l'AWS CLI pour créer les rôles et autorisations requis pour les Fenêtres de maintenance.

Rubriques

- Tâche 1 : (Facultatif) Créer un rôle de service personnalisé pour les fenêtres de maintenance (AWS CLI) (p. 697)
- Tâche 2 : Configurer des autorisations pour les utilisateurs autorisés à enregistrer des tâches de fenêtre de maintenance (AWS CLI) (p. 699)
- Tâche 3 : Configurer des autorisations pour les utilisateurs qui ne sont pas autorisés à enregistrer des tâches de fenêtre de maintenance (AWS CLI) (p. 702)

Tâche 1 : (Facultatif) Créer un rôle de service personnalisé pour les fenêtres de maintenance (AWS CLI)

> **Important**
>
> Un rôle de service personnalisé n'est pas requis si vous choisissez d'utiliser un rôle lié à un service Systems Manager pour laisser les fenêtres de maintenance exécuter des tâches en votre nom. Si vous n'avez pas de rôle lié à un service Systems Manager dans votre compte, vous pouvez le créer lorsque vous créez ou mettez à jour une tâche de fenêtre de maintenance à l'aide de la console Systems Manager. Pour de plus amples informations, consultez les rubriques suivantes :
>
> - Dois-je utiliser un rôle lié à un service ou un rôle de service personnalisé pour exécuter des tâches de fenêtre de maintenance ? (p. 691)
> - Utilisation des rôles liés à un service pour Systems Manager (p. 1312)
> - Attribuer des tâches à une fenêtre de maintenance (console) (p. 712)

1. Copiez et collez la stratégie d'approbation suivante dans un fichier texte. Enregistrez le fichier avec le nom et l'extension de fichier suivants : `mw-role-trust-policy.json`.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "ssm.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

2. Ouvrez l' AWS CLI et exécutez la commande suivante dans le répertoire où vous avez placé afin de créer un rôle de fenêtre de maintenance appelé `mw-role-trust-policy.json` .`my-maintenance-window-role` La commande attribue la stratégie que vous avez créée à l'étape précédente à ce rôle.

   Linux

```
aws iam create-role \
    --role-name "my-maintenance-window-role" \
    --assume-role-policy-document file://mw-role-trust-policy.json
```

   Windows

```
aws iam create-role ^
    --role-name "my-maintenance-window-role" ^
    --assume-role-policy-document file://mw-role-trust-policy.json
```

   Le système renvoie des informations similaires à celles qui suivent.

```
{
    "Role": {
        "AssumeRolePolicyDocument": {
            "Version": "2012-10-17",
            "Statement": [
                {
                    "Action": "sts:AssumeRole",
                    "Effect": "Allow",
                    "Principal": {
                        "Service": "ssm.amazonaws.com"
                    }
                }
            ]
        },
        "RoleId": "AROAIIZKPBKS2LEXAMPLE",
        "CreateDate": "2017-04-04T03:40:17.373Z",
        "RoleName": "my-maintenance-window-role",
        "Path": "/",
        "Arn": "arn:aws:iam::123456789012:role/my-maintenance-window-role"
    }
}
```

Note

Notez les valeurs de `RoleName` et `Arn`. Vous les spécifiez lorsque vous créez une fenêtre de maintenance qui utilise ce rôle personnalisé.

3. Exécutez la commande suivante pour attacher la stratégie gérée `AmazonSSMMaintenanceWindowRole` au rôle que vous avez créé lors de l'étape 2.

   Linux

```
aws iam attach-role-policy \
    --role-name "my-maintenance-window-role" \
    --policy-arn "arn:aws:iam::aws:policy/service-role/
AmazonSSMMaintenanceWindowRole"
```

   Windows

```
aws iam attach-role-policy ^
    --role-name "my-maintenance-window-role" ^
    --policy-arn "arn:aws:iam::aws:policy/service-role/
AmazonSSMMaintenanceWindowRole"
```

Tâche 2 : Configurer des autorisations pour les utilisateurs autorisés à enregistrer des tâches de fenêtre de maintenance (AWS CLI)

Lorsque vous enregistrez une tâche avec une fenêtre de maintenance, vous spécifiez un rôle de service personnalisé ou un rôle lié à un service Systems Manager pour exécuter les opérations de tâche réelles. Il s'agit du rôle assumé par le service lorsqu'il exécutera des tâches en votre nom. Avant cela, pour enregistrer la tâche elle-même, vous devez affecter la stratégie IAM PassRole à un compte utilisateur IAM ou à un groupe IAM. Cela permet à l'utilisateur IAM ou au groupe IAM de spécifier, dans le cadre de l'enregistrement de ces tâches dans la fenêtre de maintenance, le rôle à utiliser lors de l'exécution des tâches. Pour de plus amples informations, veuillez consulter Octroi d'autorisations à un utilisateur pour transférer un rôle à un service AWS dans le IAM Guide de l'utilisateur.

Pour configurer des autorisations pour les utilisateurs autorisés à enregistrer des tâches de fenêtre de maintenance (AWS CLI)

1. Copiez et collez la IAM stratégie suivante dans un éditeur de texte et enregistrez-la avec le nom et l'extension de fichier suivants : `mw-passrole-policy.json`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "custom-role-arn"
        },
        {
            "Effect": "Allow",
            "Action": "iam:ListRoles",
            "Resource": "arn:aws:iam::account-id:role/"
        },
        {
            "Effect": "Allow",
            "Action": "iam:ListRoles",
            "Resource": "arn:aws:iam::account-id:role/aws-service-role/
ssm.amazonaws.com/"
```

```
        }
    ]
}
```

Remplacez *custom-role-arn* avec l'ARN du rôle de fenêtre de maintenance personnalisé que vous avez créé précédemment, par exemple `arn:aws:iam::123456789012:role/my-maintenance-window-role` .

Remplacez *account-id* dans les deux `iam:ListRoles` autorisations avec l'ID de votre compte AWS. L'ajout de cette autorisation pour la ressource `arn:aws:iam::account-id:role/` permet aux utilisateurs du groupe d'afficher et de choisir parmi les rôles clients dans la console lorsqu'ils créent une tâche de fenêtre de maintenance. L'ajout de cette autorisation pour `arn:aws:iam::account-id:role/aws-service-role/ssm.amazonaws.com/` permet aux utilisateurs du groupe de choisir le rôle lié au service Systems Manager dans la console lorsqu'ils créent une tâche de fenêtre de maintenance.

2. Ouvrez AWS CLI.
3. Selon que vous voulez attribuer l'autorisation à un utilisateur ou à un groupe IAM, exécutez l'une des commandes suivantes.

   - Pour un utilisateur IAM :

     Linux

     ```
     aws iam put-user-policy \
         --user-name "user-name" \
         --policy-name "policy-name" \
         --policy-document file://path-to-document
     ```

     Windows

     ```
     aws iam put-user-policy ^
         --user-name "user-name" ^
         --policy-name "policy-name" ^
         --policy-document file://path-to-document
     ```

     Dans *user-name*, spécifiez lIAM'utilisateur qui attribue des tâches aux fenêtres de maintenance. Dans *policy-name*, spécifiez le nom que vous souhaitez utiliser pour identifier la stratégie, par exemple **`my-iam-passrole-policy`** . Pour *path-to-document*, spécifiez le chemin d'accès au fichier que vous avez enregistré à l'étape 1. Par exemple: `file://C:\Temp\mw-passrole-policy.json`

     > **Note**
     >
     > Pour autoriser un utilisateur à enregistrer des tâches pour les fenêtres de maintenance à l'aide de la console AWS Systems Manager, vous devez également affecter la stratégie `AmazonSSMFullAccess` à votre compte d'utilisateur (ou une stratégie IAM qui fournit un ensemble plus petit d'autorisations d'accès pour Systems Manager qui couvre les tâches de la fenêtre de maintenance. Pour de plus amples informations, veuillez consulter Créer des groupes d'utilisateurs (p. 31) et Créer des utilisateurs et attribuer des autorisations (p. 32). Exécutez la commande suivante pour attribuer la `AmazonSSMFullAccess` stratégie à votre compte.
     >
     > Linux
     >
     > ```
     > aws iam attach-user-policy \
     > --policy-arn "arn:aws:iam::aws:policy/AmazonSSMFullAccess" \
     > --user-name "user-name"
     > ```

Windows

```
aws iam attach-user-policy ^
    --policy-arn "arn:aws:iam::aws:policy/AmazonSSMFullAccess" ^
    --user-name "user-name"
```

- Pour un groupe IAM :

  Linux

```
aws iam put-group-policy \
    --group-name "group-name" \
    --policy-name "policy-name" \
    --policy-document file://path-to-document
```

  Windows

```
aws iam put-group-policy ^
    --group-name "group-name" ^
    --policy-name "policy-name" ^
    --policy-document file://path-to-document
```

  Dans *group-name*, spécifiez le IAM groupe dont les membres attribuent des tâches aux fenêtres de maintenance. Dans *policy-name*, spécifiez le nom que vous souhaitez utiliser pour identifier la stratégie, par exemple **my-iam-passrole-policy** . Pour *path-to-document*, spécifiez le chemin d'accès au fichier que vous avez enregistré à l'étape 1. Par exemple: `file://C:\Temp\mw-passrole-policy.json`

  Note

  Pour autoriser les membres d'un groupe à enregistrer des tâches pour les fenêtres de maintenance à l'aide de la console AWS Systems Manager, vous devez également affecter la stratégie `AmazonSSMFullAccess` à votre groupe. Exécutez la commande suivante pour attribuer cette stratégie à votre groupe.

  Linux

```
aws iam attach-group-policy \
    --policy-arn "arn:aws:iam::aws:policy/AmazonSSMFullAccess" \
    --group-name "group-name"
```

  Windows

```
aws iam attach-group-policy ^
    --policy-arn "arn:aws:iam::aws:policy/AmazonSSMFullAccess" ^
    --group-name "group-name"
```

4. Exécutez la commande suivante pour vérifier que la stratégie a bien été attribuée au groupe.

   Linux

```
aws iam list-group-policies \
    --group-name "group-name"
```

   Windows

```
aws iam list-group-policies ^
```

```
    --group-name "group-name"
```

Tâche 3 : Configurer des autorisations pour les utilisateurs qui ne sont pas autorisés à enregistrer des tâches de fenêtre de maintenance (AWS CLI)

1. Copiez et collez la IAM stratégie suivante dans un éditeur de texte et enregistrez-la avec le nom et l'extension de fichier suivants : `deny-mw-tasks-policy.json`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": "ssm:RegisterTaskWithMaintenanceWindow",
            "Resource": "*"
        }
    ]
}
```

2. Ouvrez AWS CLI.
3. Selon que vous voulez attribuer l'autorisation à un utilisateur ou à un groupe IAM, exécutez l'une des commandes suivantes.

   - Pour un utilisateur IAM :

     Linux

     ```
     aws iam put-user-policy \
         --user-name "user-name" \
         --policy-name "policy-name" \
         --policy-document file://path-to-document
     ```

     Windows

     ```
     aws iam put-user-policy ^
         --user-name "user-name" ^
         --policy-name "policy-name" ^
         --policy-document file://path-to-document
     ```

     Dans *user-name*, spécifiez lIAM'utilisateur pour empêcher d'affecter des tâches aux fenêtres de maintenance. Dans *policy-name*, spécifiez le nom que vous souhaitez utiliser pour identifier la stratégie, par exemple **my-deny-mw-tasks-policy** . Pour *path-to-document*, spécifiez le chemin d'accès au fichier que vous avez enregistré à l'étape 1. Par exemple: `file://C:\Temp\deny-mw-tasks-policy.json`

   - Pour un groupe IAM :

     Linux

     ```
     aws iam put-group-policy \
         --group-name "group-name" \
         --policy-name "policy-name" \
         --policy-document file://path-to-document
     ```

     Windows

     ```
     aws iam put-group-policy ^
         --group-name "group-name" ^
```

```
    --policy-name "policy-name" ^
    --policy-document file://path-to-document
```

Dans *group-name*, spécifiez le IAM groupe dont vous souhaitez empêcher l'affectation de tâches aux fenêtres de maintenance. Dans *policy-name*, spécifiez le nom que vous souhaitez utiliser pour identifier la stratégie, par exemple **my-deny-mw-tasks-policy** . Pour *path-to-document*, spécifiez le chemin d'accès au fichier que vous avez enregistré à l'étape 1. Par exemple: `file://C:\Temp\deny-mw-tasks-policy.json`

4. Exécutez la commande suivante pour vérifier que la stratégie a bien été attribuée au groupe.

   Linux

```
aws iam list-group-policies \
    --group-name "group-name"
```

   Windows

```
aws iam list-group-policies ^
    --group-name "group-name"
```

Contrôler l'accès aux fenêtres de maintenance (Outils pour Windows PowerShell)

Les procédures suivantes expliquent comment utiliser les Outils pour Windows PowerShell pour créer les rôles et autorisations requis pour la fonctionnalité de Fenêtres de maintenance.

Rubriques

- Tâche 1 : (Facultatif) Créez un rôle de service personnalisé pour les fenêtres de maintenance (Outils pour Windows PowerShell) (p. 703)
- Tâche 2 : Configurer des autorisations pour les utilisateurs autorisés à enregistrer des tâches de fenêtre de maintenance (PowerShell) (p. 704)
- Tâche 3 : Configurer des autorisations pour les utilisateurs qui ne sont pas autorisés à enregistrer des tâches de fenêtre de maintenance (PowerShell) (p. 706)

Tâche 1 : (Facultatif) Créez un rôle de service personnalisé pour les fenêtres de maintenance (Outils pour Windows PowerShell)

> **Important**
>
> Un rôle de service personnalisé n'est pas requis si vous choisissez d'utiliser un rôle lié à un service Systems Manager pour laisser les fenêtres de maintenance exécuter des tâches en votre nom. Si vous n'avez pas de rôle lié à un service Systems Manager dans votre compte, vous pouvez le créer lorsque vous créez ou mettez à jour une tâche de fenêtre de maintenance à l'aide de la console Systems Manager. Pour de plus amples informations, consultez les rubriques suivantes :
>
> - Dois-je utiliser un rôle lié à un service ou un rôle de service personnalisé pour exécuter des tâches de fenêtre de maintenance ? (p. 691)
> - Utilisation des rôles liés à un service pour Systems Manager (p. 1312)
> - Attribuer des tâches à une fenêtre de maintenance (console) (p. 712)

1. Copiez et collez la stratégie d'approbation suivante dans un fichier texte. Enregistrez le fichier avec le nom et l'extension de fichier suivants : `mw-role-trust-policy.json`.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "ssm.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

2. Ouvrez Outils pour Windows PowerShell et exécutez la commande suivante pour créer un rôle avec un nom qui identifie ce rôle en tant que rôle de fenêtre de maintenance ; par exemple `my-maintenance-window-role` . Le rôle utilise la stratégie que vous avez créée à l'étape précédente.

```
New-IAMRole `
    -RoleName "my-maintenance-window-role" `
    -AssumeRolePolicyDocument (Get-Content -raw .\mw-role-trust-policy.json)
```

Le système renvoie des informations similaires à celles qui suivent.

```
Arn : arn:aws:iam::123456789012:role/mw-task-role
AssumeRolePolicyDocument : ExampleDoc12345678
CreateDate : 4/4/2017 11:24:43
Path : /
RoleId : AROAIIZKPBKS2LEXAMPLE
RoleName : my-maintenance-window-role
```

3. Exécutez la commande suivante pour attacher la stratégie gérée `AmazonSSMMaintenanceWindowRole` au rôle que vous avez créé à l'étape précédente.

```
Register-IAMRolePolicy `
    -RoleName "my-maintenance-window-role" `
    -PolicyArn "arn:aws:iam::aws:policy/service-role/AmazonSSMMaintenanceWindowRole"
```

Tâche 2 : Configurer des autorisations pour les utilisateurs autorisés à enregistrer des tâches de fenêtre de maintenance (PowerShell)

Lorsque vous enregistrez une tâche avec une fenêtre de maintenance, vous spécifiez un rôle de service personnalisé ou un rôle lié à un service Systems Manager pour exécuter les opérations de tâche réelles. Il s'agit du rôle assumé par le service lorsqu'il exécutera des tâches en votre nom. Avant cela, pour enregistrer la tâche elle-même, vous devez affecter la stratégie IAM PassRole à un compte utilisateur IAM ou à un groupe IAM. Cela permet à l'utilisateur IAM ou au groupe IAM de spécifier, dans le cadre de l'enregistrement de ces tâches dans la fenêtre de maintenance, le rôle à utiliser lors de l'exécution des tâches. Pour de plus amples informations, veuillez consulter Octroi d'autorisations à un utilisateur pour transférer un rôle à un service AWS dans le IAM Guide de l'utilisateur.

1. Copiez et collez la IAM stratégie suivante dans un éditeur de texte et enregistrez-la avec le nom et l'extension de fichier suivants : `mw-passrole-policy.json`.

```
{
    "Version": "2012-10-17",
```

```
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "custom-role-arn"
        },
        {
            "Effect": "Allow",
            "Action": "iam:ListRoles",
            "Resource": "arn:aws:iam::account-id:role/"
        },
        {
            "Effect": "Allow",
            "Action": "iam:ListRoles",
            "Resource": "arn:aws:iam::account-id:role/aws-service-role/
ssm.amazonaws.com/"
        }
    ]
}
```

Remplacez *custom-role-arn* avec l'ARN du rôle de fenêtre de maintenance personnalisé que vous avez créé précédemment, par exemple `arn:aws:iam::123456789012:role/my-maintenance-window-role` .

Remplacez *account-id* dans les deux `iam:ListRoles` autorisations avec l'ID de votre compte AWS. L'ajout de cette autorisation pour la ressource `arn:aws:iam::`*account-id*`:role/` permet aux utilisateurs du groupe d'afficher et de choisir parmi les rôles clients dans la console lorsqu'ils créent une tâche de fenêtre de maintenance. L'ajout de cette autorisation pour `arn:aws:iam::`*account-id*`:role/aws-service-role/ssm.amazonaws.com/` permet aux utilisateurs du groupe de choisir le rôle lié au service Systems Manager dans la console lorsqu'ils créent une tâche de fenêtre de maintenance.

2. Ouvrez Outils pour Windows PowerShell.
3. Selon que vous voulez attribuer l'autorisation à un utilisateur ou à un groupe IAM, exécutez l'une des commandes suivantes.

   - Pour un utilisateur IAM :

```
Write-IAMUserPolicy `
    -UserName "user-name" `
    -PolicyDocument (Get-Content -raw path-to-document) `
    -PolicyName "policy-name"
```

   Dans *user-name*, spécifiez lIAM'utilisateur qui attribue des tâches aux fenêtres de maintenance. Dans *policy-name*, spécifiez le nom que vous souhaitez utiliser pour identifier la stratégie, par exemple **my-iam-passrole-policy** . Pour *path-to-document*, spécifiez le chemin d'accès au fichier que vous avez enregistré à l'étape 1. Par exemple: `C:\temp\passrole-policy.json`

   > **Note**
   >
   > Si vous prévoyez d'enregistrer des tâches pour des fenêtres de maintenance à l'aide de la console AWS Systems Manager, vous devez aussi attribuer la stratégie `AmazonSSMFullAccess` à votre compte d'utilisateur. Exécutez la commande suivante pour attribuer cette stratégie à votre compte.

```
Register-IAMUserPolicy `
    -UserName "user-name" `
    -PolicyArn "arn:aws:iam::aws:policy/AmazonSSMFullAccess"
```

   - Pour un groupe IAM :

```
Write-IAMGroupPolicy `
    -GroupName "group-name" `
    -PolicyDocument (Get-Content -raw path-to-document) `
    -PolicyName "policy-name"
```

Dans *group-name*, spécifiez le IAM groupe qui attribue les tâches aux fenêtres de maintenance. Dans *policy-name*, spécifiez le nom que vous souhaitez utiliser pour identifier la stratégie, par exemple **my-iam-passrole-policy** . Pour *path-to-document*, spécifiez le chemin d'accès au fichier que vous avez enregistré à l'étape 1. Par exemple: `C:\temp\passrole-policy.json`

> **Note**
>
> Si vous prévoyez d'enregistrer des tâches pour des fenêtres de maintenance à l'aide de la console AWS Systems Manager, vous devez aussi attribuer la stratégie `AmazonSSMFullAccess` à votre compte d'utilisateur. Exécutez la commande suivante pour attribuer cette stratégie à votre groupe.

```
Register-IAMGroupPolicy `
    -GroupName "group-name" `
    -PolicyArn "arn:aws:iam::aws:policy/AmazonSSMFullAccess"
```

4. Exécutez la commande suivante pour vérifier que la stratégie a bien été attribuée au groupe.

```
Get-IAMGroupPolicies `
    -GroupName "group-name"
```

Tâche 3 : Configurer des autorisations pour les utilisateurs qui ne sont pas autorisés à enregistrer des tâches de fenêtre de maintenance (PowerShell)

1. Copiez et collez la IAM stratégie suivante dans un éditeur de texte et enregistrez-la avec le nom et l'extension de fichier suivants : `deny-mw-tasks-policy.json`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": "ssm:RegisterTaskWithMaintenanceWindow",
            "Resource": "*"
        }
    ]
}
```

2. Ouvrez Outils pour Windows PowerShell.
3. Selon que vous voulez attribuer l'autorisation à un utilisateur ou à un groupe IAM, exécutez l'une des commandes suivantes.
   - Pour un utilisateur IAM :

```
Write-IAMUserPolicy `
    -UserName "user-name" `
    -PolicyDocument (Get-Content -raw path-to-document) `
    -PolicyName "policy-name"
```

Dans *user-name*, spécifiez lIAM'utilisateur pour empêcher d'affecter des tâches aux fenêtres de maintenance. Dans *policy-name*, spécifiez le nom que vous souhaitez utiliser pour identifier la stratégie, par exemple **my-deny-mw-tasks-policy** . Pour *path-to-document*, spécifiez le chemin d'accès au fichier que vous avez enregistré à l'étape 1. Par exemple: `C:\temp\deny-mw-tasks-policy.json`

- Pour un groupe IAM :

```
Write-IAMGroupPolicy `
    -GroupName "group-name" `
    -PolicyDocument (Get-Content -raw path-to-document) `
    -PolicyName "policy-name"
```

Dans *group-name*, spécifiez le IAM groupe pour empêcher d'affecter des tâches aux fenêtres de maintenance. Dans *policy-name*, spécifiez le nom que vous souhaitez utiliser pour identifier la stratégie, par exemple **my-deny-mw-tasks-policy** . Pour *path-to-document*, spécifiez le chemin d'accès au fichier que vous avez enregistré à l'étape 1. Par exemple: `C:\temp\deny-mw-tasks-policy.json`

4. Exécutez la commande suivante pour vérifier que la stratégie a bien été attribuée au groupe.

```
Get-IAMGroupPolicies `
    -GroupName "group-name"
```

Utilisation des fenêtres de maintenance (console)

Cette section indique comment créer, configurer, mettre à jour et supprimer des fenêtres de maintenance à l'aide de la console AWS Systems Manager. Cette section fournit également des informations sur la gestion des cibles et des tâches d'une fenêtre de maintenance.

> **Important**
>
> Nous vous recommandons de créer et de configurer au préalable des fenêtres de maintenance dans un environnement de test.

Avant de commencer

Avant de créer une fenêtre de maintenance, vous devez configurer l'accès aux Fenêtres de maintenance. Pour plus d'informations, consultez la section Configuration d'Fenêtres de maintenance (p. 690).

Rubriques

- Créer une fenêtre de maintenance (console) (p. 707)
- Affecter des cibles à une fenêtre de maintenance (console) (p. 709)
- Attribuer des tâches à une fenêtre de maintenance (console) (p. 712)
- Mettre à jour ou supprimer une fenêtre de maintenance (console) (p. 713)

Créer une fenêtre de maintenance (console)

Dans cette procédure, vous créez une fenêtre de maintenance et spécifiez ses options de base, telles que le nom, la planification et la durée. Dans les étapes ultérieures, vous allez choisir les cibles, ou les ressources, qu'elle met à jour, ainsi que les tâches qui s'exécutent lors de l'exécution de cette fenêtre de maintenance.

Note

Pour obtenir une explication des relations entre les différentes options liées à la planification pour les fenêtres de maintenance, consultez Référence : Options de planification de la fenêtre de maintenance et de période active (p. 772).
Pour de plus amples informations sur l'utilisation de l'option `--schedule`, veuillez consulter Référence : Expressions cron et rate pour Systems Manager (p. 1392).

Créer une fenêtre de maintenance (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fenêtres de maintenance.
3. Choisissez Create a Maintenance Window (Créer une fenêtre de maintenance).
4. Pour Name (Nom), tapez un nom descriptif pour vous aider à identifier cette fenêtre de maintenance comme fenêtre de maintenance de test.
5. Pour Description, entrez une description.
6. Choisissez Allow unregistered targets (Autoriser les cibles non enregistrées) si vous souhaitez autoriser une tâche de fenêtre de maintenance à exécuter des instances gérées, même si vous n'avez pas enregistré ces instances en tant que cibles. Si vous choisissez cette option, vous pouvez choisir des instances non enregistrées (par ID d'instance) lorsque vous enregistrez une tâche avec la fenêtre de maintenance.

   Si vous ne choisissez pas cette option, vous devez choisir des cibles enregistrées au préalable lorsque vous enregistrez une tâche avec la fenêtre de maintenance.
7. Spécifiez un programme pour la fenêtre de maintenance à l'aide d'une des trois options de programmation.

   Pour plus d'informations sur la génération d'expressions cron/rate, consultez Référence : Expressions cron et rate pour Systems Manager (p. 1392).
8. Pour Durée, entrez le nombre d'heures pendant lequel la fenêtre de maintenance devra s'exécuter. La valeur que vous spécifiez détermine l'heure de fin de la fenêtre de maintenance en fonction de l'heure de démarrage. Aucune tâche de fenêtre de maintenance n'est autorisée à démarrer après l'heure de fin résultante moins le nombre d'heures que vous spécifiez pour Stop initiating tasks (Arrêt de l'initialisation de tâches) à l'étape suivante.

   Par exemple, si la fenêtre de maintenance commence à 15 h, que la durée est de trois heures et que la valeur de Stop initiating tasks (Arrêt de l'initialisation de tâches) est d'une heure, aucune tâche de fenêtre de maintenance ne peut commencer après 17 h.
9. Dans le champ Stop initiating tasks (Arrêter le lancement des tâches), entrez le nombre d'heures avant la fin de la fenêtre de maintenance pendant lequel le système doit cesser de planifier l'exécution de nouvelles tâches.
10. (Facultatif) Pour Start date (optional) [Date de début (facultatif)], spécifiez la date et l'heure, au format ISO-8601 étendu, où vous voulez que la fenêtre de maintenance devienne active. Cela vous permet de retarder l'activation de la fenêtre de maintenance jusqu'à la date ultérieure spécifiée.
11. (Facultatif) Pour Start date (optional) [Date de début (facultatif)], spécifiez la date et l'heure, au format ISO-8601 étendu, où vous voulez que la fenêtre de maintenance devienne inactive. Cela vous permet de définir une date et une heure futures après lesquelles la fenêtre de maintenance ne s'exécutera plus.
12. (Facultatif) Pour Time zone (optional) [Fuseau horaire (facultatif)], spécifiez le fuseau horaire sur lequel baser les exécutions de fenêtre de maintenance planifiées, au format IANA (Internet Assigned Numbers Authority). Exemple : « Amérique/Los_Angeles », « etc/UTC » ou « Asie/Séoul ».

    Pour plus d'informations sur les formats valides, consultez Time Zone Database sur le site web de l'IANA.

13. (Facultatif) Dans la zone Manage tags (Gérer les balises), appliquez une ou plusieurs paires nom/valeur de clé de balise à la fenêtre de maintenance.

    Les balises sont des métadonnées facultatives que vous affectez à une ressource. Les balises vous permettent de classer une ressource de différentes façons, par exemple, par objectif, par propriétaire ou par environnement. Par exemple, vous pouvez baliser une fenêtre de maintenance pour identifier le type de tâches qu'elle exécute, les types de cibles et l'environnement dans lequel elle s'exécute. Dans ce cas, vous pouvez spécifier les paires nom/valeur de clé suivantes :

    - `Key=TaskType,Value=AgentUpdate`
    - `Key=OS,Value=Windows`
    - `Key=Environment,Value=Production`
14. Choisissez Create maintenance window (Créer une fenêtre de maintenance). Le système vous renvoie à la page de la fenêtre de maintenance. L'état de la fenêtre de maintenance que vous venez de créer est Enabled (Activé).

Affecter des cibles à une fenêtre de maintenance (console)

Dans cette procédure, vous pouvez enregistrer une cible avec une fenêtre de maintenance. En d'autres termes, vous spécifiez sur quelles ressources la fenêtre de maintenance effectue des actions.

Affecter des cibles à une fenêtre de maintenance (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fenêtres de maintenance.
3. Dans la liste des fenêtres de maintenance, sélectionnez la fenêtre de maintenance à laquelle ajouter les cibles.
4. Choisissez Actions, puis Register targets (Enregistrer les cibles).
5. (Facultatif) Pour Target Name (Nom de cible), saisissez un nom pour les cibles.
6. (Facultatif) Sous Description, entrez une description.
7. (Facultatif) Pour Owner information (Informations sur le propriétaire), spécifiez les informations à inclure dans tout événement déclenché lors de l'exécution de tâches pour ces cibles dans cette fenêtre de maintenance.EventBridge

    Pour de plus amples informations sur l'utilisation de EventBridge pour surveiller les événements Systems Manager, veuillez consulter Surveillance des événements Systems Manager avec Amazon EventBridge (p. 1340).
8. Dans la zone Targets (Cibles), choisissez l'une des options décrites dans le tableau suivant.

Option	Description
Spécification de balises d'instance	Pour les Instance tags (Balises d'instance), spécifiez un ou plusieurs clés de balise et (facultatif) des valeurs de balise qui ont été ou seront ajoutées aux instances gérées dans votre compte. Lorsque la fenêtre de maintenance s'exécute, elle tente d'exécuter des tâches sur toutes les instances gérées auxquelles ces balises ont été ajoutées. Si vous spécifiez plusieurs clés de balise, une instance doit être balisée avec toutes les clés

Option	Description
	et valeurs de balise que vous décidez d'inclure dans le groupe cible.
Choix manuel des instances	Dans la liste, cochez la case en regard de chaque instance que vous souhaitez inclure dans la fenêtre de maintenance cible. La liste inclut toutes les instances dans votre compte qui sont configurées pour être utilisées avec Systems Manager. Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage. Pour les instances sur site et les machines virtuelles (VM), consultez Configuration d'AWS Systems Manager pour les environnements hybrides (p. 49)

Option	Description
Choix de groupes de ressources	Pour Resource group (Groupe de ressources), choisissez dans la liste le nom d'un groupe de ressources existant dans votre compte. Pour plus d'informations sur la création et l'utilisation de groupes de ressources, consultez les rubriques suivantes : • Présentation d'Groupes de ressources AWS dans le Guide de l'utilisateur Groupes de ressources AWS • Groupes de ressources et balisage pour AWS dans le blog AWS News Blog. Pour Resource types (Types de ressource), sélectionnez jusqu'à cinq types de ressources disponibles, ou choisissez All resource types (Tous les types de ressource). Si les tâches que vous attribuez à la fenêtre de maintenance n'agissent pas sur l'un des types de ressources que vous avez ajoutés à la cible, le système peut signaler une erreur. Les tâches pour lesquelles un type de ressource pris en charge est trouvé continuent de s'exécuter malgré ces erreurs. Par exemple, supposons que vous ajoutez les types de ressource suivants pour cette cible : • `AWS::S3::Bucket` • `AWS::DynamoDB::Table` • `AWS::EC2::Instance` Mais plus tard, lorsque vous ajoutez des tâches à la fenêtre de maintenance, vous incluez uniquement des tâches qui effectuent des actions sur les instances, comme l'application d'une référence de correctifs ou le redémarrage d'une instance. Dans le journal maintenancewindow, une erreur peut être signalée si aucun compartiment ou table Amazon Simple Storage Service (Amazon S3) n'est trouvé.Amazon DynamoDB Toutefois, la fenêtre de maintenance continue d'exécuter des tâches sur les instances dans votre groupe de ressources.

9. Choisissez Register targets (Enregistrer les cibles).

Si vous souhaitez attribuer d'autres cibles à cette fenêtre de maintenance, choisissez l'onglet Targets (Cibles), puis choisissez Register targets (Enregistrer des cibles). Avec cette option, vous pouvez choisir d'autres méthodes de ciblage. Par exemple, si vous avez ciblé au préalable des instances par

ID d'instance, vous pouvez enregistrer de nouvelles cibles et instances de cible en spécifiant les balises appliquées aux instances gérées ou en choisissant des types de ressource dans un groupe de ressources..

Attribuer des tâches à une fenêtre de maintenance (console)

Dans cette procédure, vous ajoutez une tâche à une fenêtre de maintenance. Les tâches sont les actions effectuées sur une ressource au cours de l'exécution d'une fenêtre de maintenance.

Les quatre types de tâches peuvent être ajoutés à une fenêtre de maintenance :

- Commandes de Fonctionnalité Exécuter la commande AWS Systems Manager
- Flux de travail d'automatisation Systems Manager
- Fonctions AWS Lambda
- Tâches AWS Step Functions

Pour attribuer des tâches à une fenêtre de maintenance

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fenêtres de maintenance.
3. Dans la liste des fenêtres de maintenance, choisissez une fenêtre de maintenance.
4. Choisissez Actions, puis choisissez l'option pour le type de tâche que vous voulez enregistrer avec la fenêtre de maintenance :
   - Enregistrer une tâche d'exécution de commande
   - Enregistrer la tâche d'automatisation
   - Enregistrer une tâche Lambda
   - Enregistrer la tâche Step Functions
5. Pour Name (Nom), entrez le nom de la tâche.
6. Pour Description, entrez une description.
7. Pour Document, sélectionnez le document de commande SSM ou Automation qui définit les tâches à exécuter.
8. Pour Version du document (pour les tâches Automation), choisissez la version du document à utiliser.
9. Pour Priorité de tâche, spécifiez la priorité de cette tâche. Zéro (`0`) est la priorité la plus élevée. Les tâches d'une fenêtre de maintenance sont planifiées par ordre de priorité des tâches qui ont la même priorité planifiée en parallèle.
10. Dans la section Cibles, identifiez les instances sur lesquelles vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant manuellement les instances ou en indiquant un groupe de ressources.

    > **Note**
    >
    > Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.
11. (Facultatif) Pour Contrôle du débit :
    - Pour Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage d'instances sur lesquelles exécuter simultanément la commande.

      > **Note**
      >
      > Si vous avez sélectionné des cibles en indiquant des balises appliquées à des instances gérées ou en indiquant des groupes de ressources AWS, et que vous n'êtes pas sûr du

nombre d'instances ciblées, limitez le nombre d'instances pouvant exécuter simultanément le document en spécifiant un pourcentage.

- Pour Seuil d'erreur, spécifiez quand arrêter l'exécution de la commande sur d'autres instances après son échec sur un nombre ou un pourcentage d'instances. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les instances sur lesquelles la commande est en cours de traitement peuvent également envoyer des erreurs.

12. Dans la zone IAM service role (Rôle de service IAM), choisissez l'une des options suivantes pour autoriser Systems Manager à exécuter des tâches sur vos instances cibles :

    - Créer et utiliser un rôle lié à un service pour Systems Manager

      Les rôles liés à un service offrent une manière sécurisée d'accorder des autorisations aux services AWS, car seul le service lié peut assumer un rôle lié à un service. De plus, AWS définit automatiquement les autorisations des rôles liés à un service, en fonction des actions que le service lié effectue en votre nom.

      Note

      Si un rôle lié à un service a déjà été créé pour votre compte, choisissez Use the service-linked role for Systems Manager (Utiliser le rôle lié à un service pour Systems Manager).

    - Use a custom service role (Utiliser un rôle de service personnalisé)

      Vous pouvez créer un rôle de service personnalisé pour les tâches de fenêtre de maintenance si vous souhaitez utiliser des autorisations plus rigoureuses que celles fournies par le rôle lié à un service.

      Si vous avez besoin de créer un rôle de service personnalisé, veuillez consulter l'une des rubriques suivantes :

      - Contrôle de l'accès aux fenêtres de maintenance (console) (p. 692)
      - Contrôler l'accès aux fenêtres de maintenance (AWS CLI) (p. 697)
      - Contrôler l'accès aux fenêtres de maintenance (Outils pour Windows PowerShell) (p. 703)

    Pour choisir plus facilement d'utiliser un rôle de service personnalisé ou le rôle lié à un service Systems Manager avec une tâche de fenêtre de maintenance, consultez Dois-je utiliser un rôle lié à un service ou un rôle de service personnalisé pour exécuter des tâches de fenêtre de maintenance ? (p. 691).

13. Dans la section Input Parameters (Paramètres d'entrée), spécifiez les paramètres du document. Pour les documents Automation, le système fournit automatiquement certaines des valeurs. Vous pouvez conserver ou remplacer ces valeurs.
14. Exécutez l'assistant.

Mettre à jour ou supprimer une fenêtre de maintenance (console)

Vous pouvez mettre à jour ou supprimer une fenêtre de maintenance. Vous pouvez également mettre à jour ou supprimer les cibles ou tâches d'une fenêtre de maintenance. Si vous modifiez les détails d'une fenêtre de maintenance, vous pouvez modifier le programme, les cibles et les tâches. Vous pouvez également spécifier les noms et descriptions des fenêtres, cibles et tâches, ce qui vous aide à mieux comprendre leur utilité et facilite la gestion de votre file d'attente de fenêtres.

Cette section présente comment mettre à jour ou supprimer une fenêtre de maintenance, des cibles et des tâches à l'aide de la console AWS Systems Manager. Pour voir des exemples de la façon de procéder à l'aide de l'AWS CLI, consultez Didacticiel : Mettre à jour une fenêtre de maintenance (AWS CLI) (p. 754).

Rubriques

- Mettre à jour ou supprimer une fenêtre de maintenance (console) (p. 714)
- Mettre à jour ou supprimer les cibles de fenêtre de maintenance (console) (p. 714)
- Mettre à jour ou supprimer les tâches de fenêtre de maintenance (console) (p. 715)

Mettre à jour ou supprimer une fenêtre de maintenance (console)

Vous pouvez mettre à jour une fenêtre de maintenance pour modifier son nom, sa description et sa planification, et si la fenêtre doit autoriser les cibles non enregistrées ou non.

Pour mettre à jour ou supprimer une fenêtre de maintenance

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fenêtres de maintenance.
3. Choisissez la fenêtre de maintenance à mettre à jour ou supprimer, puis exécutez l'une des tâches suivantes :
   - Sélectionnez Supprimer. Le système vous invite à confirmer vos actions.
   - Choisissez Edit. Sur la page Edit maintenance window (Modifier la fenêtre de maintenance), modifiez les valeurs et options souhaitées, puis choisissez Edit maintenance window (Modifier la fenêtre de maintenance).

     Pour de plus amples informations sur les choix de configuration que vous pouvez effectuer, veuillez consulter Créer une fenêtre de maintenance (console) (p. 707).

Mettre à jour ou supprimer les cibles de fenêtre de maintenance (console)

Vous pouvez mettre à jour ou supprimer les cibles d'une fenêtre de maintenance. Si vous choisissez de mettre à jour une cible de fenêtre de maintenance, vous pouvez spécifier un nouveau nom, une nouvelle description ou un nouveau propriétaire. Vous pouvez également choisir différentes cibles.

Pour mettre à jour ou supprimer les cibles d'une fenêtre de maintenance

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fenêtres de maintenance.
3. Choisissez le nom de la fenêtre de maintenance à mettre à jour ou supprimer, puis exécutez l'une des tâches suivantes :
   - Pour mettre à jour des cibles, choisissez Edit (Modifier).
   - Pour supprimer des cibles, choisissez Deregister targets (Annuler l'enregistrement de cibles), puis cliquez sur l'onglet Targets (Cibles).

     Choisissez la cible à supprimer, puis cliquez sur Deregister target (Annuler l'enregistrement de la cible). Dans la fenêtre Deregister maintenance windows target (Annuler l'enregistrement de la cible de la fenêtre de maintenance), laissez l'option Safely deregister target (Annuler l'enregistrement de la cible en toute sécurité) si vous souhaitez que le système vérifie si la cible est référencée par une tâche avant de la supprimer. Si la cible est référencée par une tâche, le système renvoie une erreur et ne supprime pas la cible. Désélectionnez l'option Safely deregister target (Annuler l'enregistrement de la cible en toute sécurité) si vous souhaitez que le système supprime la cible même si elle est référencée par une tâche.

     Choisissez Deregister (Annuler l'enregistrement).

Mettre à jour ou supprimer les tâches de fenêtre de maintenance (console)

Vous pouvez mettre à jour ou supprimer les tâches d'une fenêtre de maintenance. Si vous choisissez de mettre à jour une tâche, vous pouvez spécifier un nouveau nom de tâche, une nouvelle description ou un nouveau propriétaire. Pour les tâches d'automatisation et Fonctionnalité Exécuter la commande, vous pouvez choisir un autre document SSM pour les tâches. Toutefois, vous ne pouvez pas modifier une tâche pour modifier son type. Par exemple, si vous avez créé une tâche Automation, vous ne pouvez pas modifier cette tâche et la remplacer par une tâche Fonctionnalité Exécuter la commande.

Pour mettre à jour ou supprimer les tâches d'une fenêtre de maintenance (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fenêtres de maintenance.
3. Choisissez la fenêtre de maintenance que vous souhaitez mettre à jour.
4. Choisissez l'onglet Tasks.
5. Si vous souhaitez supprimer une tâche, choisissez le bouton doté d'un petit x à côté de Modifier. Si vous souhaitez modifier la tâche, choisissez Modifier.
6. Modifiez les valeurs et options souhaitées, puis choisissez Modifier la tâche. Le système vous renvoie à la page de la fenêtre de maintenance.

Didacticiels sur les fenêtres de maintenance Systems Manager (AWS CLI)

Cette section inclut des didacticiels qui vous aident à apprendre à utiliser l'AWS Command Line Interface (AWS CLI) pour effectuer les opérations suivantes :

- Créer et configurer une fenêtre de maintenance
- Afficher des informations concernant une fenêtre de maintenance
- Afficher des informations sur les tâches des fenêtres de maintenance et l'exécution des tâches
- Mettre à jour une fenêtre de maintenance
- Supprimer une fenêtre de maintenance

Exécuter les opérations prérequises

Avant d'essayer ces didacticiels, exécutez les opérations prérequises suivantes :

- Configurez l' AWS CLI sur votre ordinateur local : Pour pouvoir exécuter AWS CLI des commandes, vous devez installer et configurer l'interface de ligne de commande sur votre ordinateur local. Pour plus d'informations, consultez la section Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).
- Vérifiez les rôles et les autorisations de la fenêtre de maintenance : Un AWS administrateur de votre compte doit vous accorder les autorisations AWS Identity and Access Management (IAM) dont vous avez besoin pour gérer les fenêtres de maintenance à l'aide de l'interface de ligne de commande. Pour plus d'informations, consultez la section Configuration d'Fenêtres de maintenance (p. 690).
- Créez ou configurez une instance compatible Systems Manager avec : Vous avez besoin d'au moins une instance EC2 configurée pour être utilisée avec afin de suivre les didacticiels.Systems Manager Cela signifie que Agent SSM est installé sur l'instance et qu'un profil IAM pour Systems Manager est attaché à l'instance.

  Nous vous recommandons de lancer une instance à partir de l'un des types d'AMI (Amazon Machine Image) suivants. Agent SSM est préinstallé sur chacun d'eux :

- Amazon Linux
- Amazon Linux 2
- AMIs optimisées pour ECS Amazon Linux 2
- Ubuntu Server 16.04, 18.04 et 20.04
- AMIs Windows Server 2008-2012 R2 publiées à partir de novembre 2016
- Windows Server 2016 et 2019

  Note

  L'Agent SSM n'est pas installé sur toutes les AMIs basées sur Amazon Linux ou Amazon Linux 2. Par exemple, l'Agent SSM n'est pas préinstallé sur les AMIs optimisées pour EKS basées sur Amazon Linux 2.

Pour plus d'informations sur l'installation de Agent SSM sur une instance, consultez les rubriques suivantes :

- Installation et configuration Agent SSM de sur les instances EC2 pourWindows Server (p. 100)
- Installation et configuration de l'Agent SSM sur les instances EC2 pour Linux (p. 75)

Pour plus d'informations sur la création et l'attachement d'un profil d'instance IAM pour Systems Manager à votre instance, consultez les rubriques suivantes :

- Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34)
- Attachement d'un profil d'instance IAM à une instance EC2 (p. 40)

- Créez des ressources supplémentaires si nécessaire : De nombreuses Fonctionnalité Exécuter la commande tâches ne nécessitent pas la création de ressources autres que celles répertoriées dans cette rubrique sur les prérequis. Pour cette raison, nous fournissons une tâche Fonctionnalité Exécuter la commande simple que vous pourrez utiliser lors de votre première utilisation des didacticiels. Vous avez également besoin d'une instance EC2 qui est configurée pour être utilisée avec Systems Manager, comme décrit ci-dessus. Une fois que vous avez configuré cette instance, vous pouvez enregistrer une simple tâche Fonctionnalité Exécuter la commande.

  La fonctionnalité Systems Manager Fenêtres de maintenance prend en charge quatre types de tâches :

  - Commandes de Fonctionnalité Exécuter la commande Systems Manager
  - Flux de travail d'automatisation Systems Manager
  - Fonctions AWS Lambda
  - Tâches AWS Step Functions

  En général, si une tâche de fenêtre de maintenance que vous souhaitez exécuter nécessite des ressources supplémentaires, vous devez les créer en premier. Par exemple, si vous souhaitez une fenêtre de maintenance qui exécute une fonction AWS Lambda, créez la fonction Lambda avant de commencer ; pour une tâche Fonctionnalité Exécuter la commande, créez le compartiment S3 dans lequel vous pouvez enregistrer la sortie de commande (si vous envisagez de le faire) ; et ainsi de suite.

Conserver une trace de la ressourceIDs

A mesure que vous effectuez les tâches de ce AWS CLI didacticiel, conservez une trace des ressources IDs générées par les commandes que vous exécutez. Vous utilisez un grand nombre de ceux-ci comme entrées pour les commandes ultérieures. Par exemple, lors de la création de la fenêtre de maintenance, le système vous fournit un ID de fenêtre de maintenance dans ce format :

```
{
   "WindowId":"mw-0c50858d01EXAMPLE"
}
```

Notez les informations suivantes générées par le systèmeIDs, car les didacticiels de cette section les utilisent :

- `WindowId`
- `WindowTargetId`
- `WindowTaskId`
- `WindowExecutionId`
- `TaskExecutionId`
- `InvocationId`
- `ExecutionId`

Vous avez également besoin de l'ID de l'instance EC2 que vous prévoyez d'utiliser dans le didacticiel. Par exemple: `i-02573cafcfEXAMPLE`

Didacticiels

- Didacticiel : Créer et configurer une fenêtre de maintenance (AWS CLI) (p. 717)
- Didacticiel : Affichage des informations sur les fenêtres de maintenance (AWS CLI) (p. 740)
- Didacticiel : Afficher les informations sur les tâches et les exécutions de tâches (AWS CLI) (p. 751)
- Didacticiel : Mettre à jour une fenêtre de maintenance (AWS CLI) (p. 754)
- Didacticiel : Supprimer une fenêtre de maintenance (AWS CLI) (p. 762)

Didacticiel : Créer et configurer une fenêtre de maintenance (AWS CLI)

Ce didacticiel montre comment utiliser l'AWS CLI pour créer et configurer une fenêtre de maintenance, ses cibles et ses tâches. Le parcours principal tout au long de ce didacticiel se compose d'étapes simples. Vous pouvez créer une fenêtre de maintenance simple, identifier une cible unique et configurer une tâche simple pour la fenêtre de maintenance à exécuter. Au fil du parcours, nous fournissons des informations que vous pouvez utiliser pour essayer des scénarios plus complexes.

Tout au long des étapes de ce didacticiel, remplacez les valeurs indiquées en italiques et en *`rouge`* par vos propres options et identifiants. Par exemple, remplacez l'ID de la fenêtre de maintenance *`mw-0c50858d01EXAMPLE`* et l'ID d' instance *`i-02573cafcfEXAMPLE`* par des ID de ressource que vous créez.

Table des matières

- Étape 1 : Créer la fenêtre de maintenance (AWS CLI) (p. 717)
- Étape 2 : Enregistrer une instance cible avec la fenêtre de maintenance (AWS CLI) (p. 719)
- Étape 3 : Enregistrer une tâche avec la fenêtre de maintenance (AWS CLI) (p. 723)

Étape 1 : Créer la fenêtre de maintenance (AWS CLI)

Au cours de cette étape, vous allez créer une fenêtre de maintenance et spécifier ses options de base, tels que le nom, la planification et la durée. Dans les étapes ultérieures, vous choisirez l'instance qu'elle met à jour et la tâche qu'elle exécute.

Dans notre exemple, vous allez créer une fenêtre de maintenance qui s'exécute toutes les cinq minutes. Normalement, vous n'exécutez pas une fenêtre de maintenance aussi fréquemment. Toutefois, cette fréquence vous permet de voir rapidement les résultats de votre didacticiel. Nous allons vous montrer comment passer à un taux moins fréquent après que la tâche a été exécutée avec succès.

Note

Pour obtenir une explication des relations entre les différentes options liées à la planification pour les fenêtres de maintenance, consultez Référence : Options de planification de la fenêtre de maintenance et de période active (p. 772).
Pour de plus amples informations sur l'utilisation de l'option `--schedule`, veuillez consulter Référence : Expressions cron et rate pour Systems Manager (p. 1392).

Pour créer une fenêtre de maintenance (AWS CLI)

1. Ouvrez l'AWS CLI et exécutez la commande suivante sur votre machine locale pour créer une fenêtre de maintenance qui effectue les opérations suivantes :

   - S'exécute toutes les cinq minutes pendant un maximum de deux heures (si nécessaire).
   - Empêche les nouvelles tâches de démarrer dans un délai d'une heure à partir de la fin de l'exécution de la fenêtre de maintenance.
   - Autorise les cibles non associées (instances que vous n'avez pas enregistrées avec la fenêtre de maintenance).
   - Indique par l'utilisation de balises personnalisées que son créateur a l'intention de l'utiliser dans un didacticiel.

   Linux

   ```
   aws ssm create-maintenance-window \
       --name "My-First-Maintenance-Window" \
       --schedule "rate(5 minutes)" \
       --duration 2 \
       --cutoff 1 \
       --allow-unassociated-targets \
       --tags "Key=Purpose,Value=Tutorial"
   ```

   Windows

   ```
   aws ssm create-maintenance-window ^
       --name "My-First-Maintenance-Window" ^
       --schedule "rate(5 minutes)" ^
       --duration 2 ^
       --cutoff 1 ^
       --allow-unassociated-targets ^
       --tags "Key"="Purpose","Value"="Tutorial"
   ```

   Le système renvoie des informations similaires à celles qui suivent :

   ```
   {
      "WindowId":"mw-0c50858d01EXAMPLE"
   }
   ```

2. Maintenant, exécutez la commande suivante pour afficher les informations détaillées sur les fenêtres de maintenance de votre compte :

   ```
   aws ssm describe-maintenance-windows
   ```

   Le système renvoie des informations similaires à celles qui suivent :

   ```
   {
   ```

```
    "WindowIdentities":[
       {
             "WindowId": "mw-0c50858d01EXAMPLE",
             "Name": "My-First-Maintenance-Window",
             "Enabled": true,
             "Duration": 2,
             "Cutoff": 1,
             "NextExecutionTime": "2019-05-11T16:46:16.991Z"
       }
    ]
}
```

Passez au Étape 2 : Enregistrer une instance cible avec la fenêtre de maintenance (AWS CLI) (p. 719).

Étape 2 : Enregistrer une instance cible avec la fenêtre de maintenance (AWS CLI)

Au cours de cette étape, vous allez enregistrer une cible avec votre nouvelle fenêtre de maintenance. Dans ce cas, vous devez spécifier l'instance à mettre à jour lorsque la fenêtre de maintenance s'exécute.

Pour obtenir un exemple d'enregistrement de plusieurs instances à la fois à l'aide de l'instance IDs , des exemples d'utilisation de balises pour identifier plusieurs instances et des exemples de spécification de groupes de ressources en tant que cibles, consultez Exemples : Enregistrement des cibles avec une fenêtre de maintenance (p. 720) .

> **Note**
>
> Vous devez déjà avoir créé une instance EC2 à utiliser dans cette étape, comme décrit dans les Prérequis du didacticiel Fenêtres de maintenance (p. 715).

Pour enregistrer une instance cible avec une fenêtre de maintenance (AWS CLI)

1. Exécutez la commande suivante sur votre machine locale :

   Linux

   ```
   aws ssm register-target-with-maintenance-window \
       --window-id "mw-0c50858d01EXAMPLE" \
       --resource-type "INSTANCE" \
       --target "Key=InstanceIds,Values=i-02573cafcfEXAMPLE"
   ```

   Windows

   ```
   aws ssm register-target-with-maintenance-window ^
       --window-id "mw-0c50858d01EXAMPLE" ^
       --resource-type "INSTANCE" ^
       --target "Key=InstanceIds,Values=i-02573cafcfEXAMPLE"
   ```

   Le système renvoie des informations similaires à celles qui suivent :

   ```
   {
      "WindowTargetId":"e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
   }
   ```

2. Maintenant, exécutez la commande suivante sur votre machine locale pour afficher les détails relatifs à la cible de votre fenêtre de maintenance :

Linux

```
aws ssm describe-maintenance-window-targets \
    --window-id "mw-0c50858d01EXAMPLE"
```

Windows

```
aws ssm describe-maintenance-window-targets ^
    --window-id "mw-0c50858d01EXAMPLE"
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
    "Targets": [
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "WindowTargetId": "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE",
            "ResourceType": "INSTANCE",
            "Targets": [
                {
                    "Key": "InstanceIds",
                    "Values": [
                        "i-02573cafcfEXAMPLE"
                    ]
                }
            ]
        }
    ]
}
```

Passez au Étape 3 : Enregistrer une tâche avec la fenêtre de maintenance (AWS CLI) (p. 723).

Exemples : Enregistrement des cibles avec une fenêtre de maintenance

Vous pouvez enregistrer une seule instance en tant que cible à l'aide de son ID d'instance, comme démontré dans Étape 2 : Enregistrer une instance cible avec la fenêtre de maintenance (AWS CLI) (p. 719). Vous pouvez également enregistrer une ou plusieurs instances en tant que cibles à l'aide des commandes présentées sur cette page.

En général, deux méthodes permettent d'identifier les instances que vous souhaitez utiliser comme cibles de fenêtre de maintenance : vous pouvez spécifier les instances individuelles ou utiliser des balises de ressources. La méthode de balises de ressources offre davantage d'options, comme illustré dans les exemples 2 à 3.

Vous pouvez également spécifier un ou plusieurs groupes de ressources en tant que cible d'une fenêtre de maintenance. Un groupe de ressources peut inclure des instances et beaucoup d'autres types de ressource AWS pris en charge. Les exemples 4 et 5 ci-après montrent comment ajouter des groupes de ressources à vos cibles de fenêtre de maintenance.

Pour plus d'informations sur la création et la gestion des groupes de ressources, consultez Qu'est-ce qu'Groupes de ressources AWS ? dans le Guide de l'utilisateur Groupes de ressources AWS et Groupes de ressources et balisage pour AWS dans le blog AWS News Blog.

Pour de plus amples informations sur les limites de la fonctionnalité Fenêtres de maintenance, en plus de celles spécifiées dans les exemples suivants, veuillez consulter Quotas de service Systems Manager dans le Référence générale d'Amazon Web Services..

Rubriques

- Exemple 1 : Enregistrer plusieurs cibles à l'aide de l'instanceIDs (p. 721)
- Exemple 2 : Enregistrement des cibles à l'aide des balises de ressource appliquées aux instances (p. 721)
- Exemple 3 : Enregistrement de cibles à l'aide d'un groupe de clés de balise (sans valeurs de balise) (p. 722)
- Exemple 4 : Enregistrer des cibles à l'aide d'un nom de groupe de ressources (p. 722)
- Exemple 5 : Enregistre les cibles en filtrant les types de ressources dans un groupe de ressources. (p. 723)

Exemple 1 : Enregistrer plusieurs cibles à l'aide de l'instanceIDs

Pour enregistrer plusieurs instances en tant que cibles à l'aide de leur instance, exécutez la commande suivante sur votre format de machine locale IDs :

Linux

```
aws ssm register-target-with-maintenance-window \
    --window-id "mw-0c50858d01EXAMPLE" \
    --resource-type "INSTANCE" \
    --target
 "Key=InstanceIds,Values=i-02573cafcfEXAMPLE,i-0471e04240EXAMPLE,i-07782c72faEXAMPLE"
```

Windows

```
aws ssm register-target-with-maintenance-window ^
    --window-id "mw-0c50858d01EXAMPLE ^
    --resource-type "INSTANCE" ^
    --target
 "Key=InstanceIds,Values=i-02573cafcfEXAMPLE,i-0471e04240EXAMPLE,i-07782c72faEXAMPLE
```

Utilisation recommandée : Ce point est particulièrement utile lorsque vous enregistrez un groupe unique d'instances avec une fenêtre de maintenance pour la première fois et qu'elles ne partagent pas une balise d'instance commune.

Quotas : Vous pouvez spécifier jusqu'à 50 instances au total pour chaque fenêtre de maintenance cible.

Exemple 2 : Enregistrement des cibles à l'aide des balises de ressource appliquées aux instances

Exécutez la commande suivante sur votre machine locale pour enregistrer les instances qui sont déjà balisées avec une paire clé-valeur que vous avez attribuée :

Linux

```
aws ssm register-target-with-maintenance-window \
    --window-id "mw-0c50858d01EXAMPLE" \
    --resource-type "INSTANCE" \
    --target "Key=tag:Region,Values=East"
```

Windows

```
aws ssm register-target-with-maintenance-window ^
    --window-id "mw-0c50858d01EXAMPLE" ^
    --resource-type "INSTANCE" ^
```

```
    --target "Key=tag:Region,Values=East"
```

Utilisation recommandée : Ce point est particulièrement utile lorsque vous enregistrez un groupe unique d'instances avec une fenêtre de maintenance pour la première fois et qu'elles partagent une balise d'instance commune.

Quotas : Vous pouvez spécifier jusqu'à cinq paires clé-valeur au total pour chaque cible. Si vous spécifiez plusieurs paires clé-valeur, une instance doit être balisée avec toutes les clés et valeurs de balise que vous spécifiez pour inclure dans le groupe cible.

> **Note**
>
> Vous pouvez baliser un groupe d'instances avec le groupe de correctifs de clé de balise et attribuer aux instances une valeur de clé commune, telle que `my-patch-group` . Gestionnaire de correctifs évalue la clé du groupe de correctifs sur les instances pour aider à déterminer quelle référence de correctifs s'applique à elles. Si votre tâche exécute le document SSM `AWS-RunPatchBaseline` (ou le document SSM `AWS-ApplyPatchBaseline` existant), vous pouvez spécifier la même paire clé-valeur Groupe de correctifs lorsque vous enregistrez les cibles avec une fenêtre de maintenance. Exemple : `--target "Key=tag:Patch Group,Values=my-patch-group`. Cela vous permet d'utiliser facilement une fenêtre de maintenance pour mettre à jour les correctifs sur un groupe d'instances qui sont déjà associées à la même référence de correctifs. Pour plus d'informations, consultez la section À propos des groupes de correctifs (p. 1105).

Exemple 3 : Enregistrement de cibles à l'aide d'un groupe de clés de balise (sans valeurs de balise)

Exécutez la commande suivante sur votre machine locale pour enregistrer les instances qui ont toutes une ou plusieurs clés de balise attribuées, indépendamment de leurs valeurs de clé.

Linux

```
aws ssm register-target-with-maintenance-window \
    --window-id "mw-0c50858d01EXAMPLE" \
    --resource-type "INSTANCE" \
    --target "Key=tag-key,Values=Name,Instance-Type,CostCenter"
```

Windows

```
aws ssm register-target-with-maintenance-window ^
    --window-id "mw-0c50858d01EXAMPLE" ^
    --resource-type "INSTANCE" ^
    --target "Key=tag-key,Values=Name,Instance-Type,CostCenter"
```

Utilisation recommandée : Utile lorsque vous souhaitez cibler des instances en spécifiant plusieurs clés de balise (sans leurs valeurs) plutôt qu'une seule clé de balise ou une paire clé-valeur de balise.

Quotas : Vous pouvez spécifier jusqu'à cinq clés de balise au total pour chaque cible. Si vous spécifiez plusieurs clés de balise, une instance doit être balisée avec toutes les clés de balise que vous spécifiez pour être incluse dans le groupe cible.

Exemple 4 : Enregistrer des cibles à l'aide d'un nom de groupe de ressources

Exécutez la commande suivante sur votre machine locale pour enregistrer un groupe de ressources spécifié, quel que soit le type des ressources que celui-ci contient. Si les tâches que vous attribuez à la fenêtre de maintenance n'agissent pas sur un type de ressource inclus dans le groupe de ressources, le système peut signaler une erreur. Les tâches pour lesquelles un type de ressource pris en charge est trouvé continuent de s'exécuter malgré ces erreurs.

Linux

```
aws ssm register-target-with-maintenance-window \
    --window-id "mw-0c50858d01EXAMPLE" \
    --resource-type "RESOURCE_GROUP" \
    --target "Key=resource-groups:Name,Values=MyResourceGroup"
```

Windows

```
aws ssm register-target-with-maintenance-window ^
    --window-id "mw-0c50858d01EXAMPLE" ^
    --resource-type "RESOURCE_GROUP" ^
    --target "Key=resource-groups:Name,Values=MyResourceGroup"
```

Utilisation recommandée : Utile lorsque vous souhaitez spécifier rapidement un groupe de ressources en tant que cible sans évaluer si tous ses types de ressources seront ciblés par une fenêtre de maintenance, ou lorsque vous savez que le groupe de ressources contient uniquement les types de ressources sur lesquels vos tâches effectuent des actions.

Quotas : Vous ne pouvez spécifier qu'un seul groupe de ressources en tant que cible.

Exemple 5 : Enregistre les cibles en filtrant les types de ressources dans un groupe de ressources.

Exécutez la commande suivante sur votre machine locale pour enregistrer uniquement certains types de ressource qui appartiennent à un groupe de ressources que vous spécifiez. Avec cette option, même si vous ajoutez une tâche pour un type de ressource qui appartient au groupe de ressources, la tâche ne s'exécute pas si vous n'avez pas ajouté explicitement ce type de ressource au filtre.

Linux

```
aws ssm register-target-with-maintenance-window \
    --window-id "mw-0c50858d01EXAMPLE" \
    --resource-type "RESOURCE_GROUP" \
    --target "Key=resource-groups:Name,Values=MyResourceGroup" \
    "Key=resource-
groups:ResourceTypeFilters,Values=AWS::EC2::Instance,AWS::ECS::Cluster"
```

Windows

```
aws ssm register-target-with-maintenance-window ^
    --window-id "mw-0c50858d01EXAMPLE" ^
    --resource-type "RESOURCE_GROUP" ^
    --target "Key=resource-groups:Name,Values=MyResourceGroup" ^
    "Key=resource-
groups:ResourceTypeFilters,Values=AWS::EC2::Instance,AWS::ECS::Cluster"
```

Utilisation recommandée : Utile lorsque vous souhaitez conserver un contrôle strict sur les types de ressources AWS sur lesquels votre fenêtre de maintenance peut exécuter des actions, ou lorsque votre groupe de ressources contient un grand nombre de types de ressources et que vous souhaitez éviter les rapports d'erreur inutiles dans vos journaux de fenêtre de maintenance.

Quotas : Vous ne pouvez spécifier qu'un seul groupe de ressources en tant que cible.

Étape 3 : Enregistrer une tâche avec la fenêtre de maintenance (AWS CLI)

Dans cette étape du didacticiel, vous enregistrez une tâche Fonctionnalité Exécuter la commande qui exécute la commande `df` sur votre instance EC2 pour Linux. Les résultats de cette commande Linux

standard montrent la quantité d'espace disponible et la quantité d'espace utilisée sur le système de fichiers du disque de votre instance.

-ou-

Si vous ciblez une instance EC2 pour Windows Server au lieu de Linux, remplacez df dans la commande suivante par ipconfig . Le résultat de cette commande répertorie les détails relatifs à l'adresse IP, au masque de sous-réseau et à la passerelle par défaut pour les cartes sur l'instance cible.

Lorsque vous êtes prêt à enregistrer d'autres types de tâches ou à utiliser plusieurs des options Fonctionnalité Exécuter la commande disponibles, consultez Exemples : Enregistrer les tâches avec une fenêtre de maintenance (p. 727). Là, nous fournissons plus d'informations sur les quatre types de tâches, et certaines de leurs principales options, pour vous aider à planifier des scénarios concrets plus vastes.

Pour enregistrer une tâche avec une fenêtre de maintenance

1. Exécutez la commande suivante sur votre machine locale. La version à exécuter à partir d'une machine Windows locale inclut les caractères d'échappement (« / ») dont vous avez besoin pour exécuter la commande à partir de votre outil de ligne de commande.

   Linux

   ```
   aws ssm register-task-with-maintenance-window \
       --window-id mw-0c50858d01EXAMPLE \
       --task-arn "AWS-RunShellScript" \
       --max-concurrency 1 --max-errors 1 \
       --priority 10 \
       --targets "Key=InstanceIds,Values=i-0471e04240EXAMPLE" \
       --task-type "RUN_COMMAND" \
       --task-invocation-parameters '{"RunCommand":{"Parameters":{"commands":
   ["df"]}}}'
   ```

   Windows

   ```
   aws ssm register-task-with-maintenance-window ^
       --window-id mw-0c50858d01EXAMPLE ^
       --task-arn "AWS-RunShellScript" ^
       --max-concurrency 1 --max-errors 1 ^
       --priority 10 ^
       --targets "Key=InstanceIds,Values=i-02573cafcfEXAMPLE" ^
       --task-type "RUN_COMMAND" ^
       --task-invocation-parameters={\"RunCommand\":{\"Parameters\":{\"commands\":
   [\"df\"]}}}
   ```

   Le système renvoie des informations similaires à ce qui suit :

   ```
   {
       "WindowTaskId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE"
   }
   ```

2. Maintenant, exécutez la commande suivante pour afficher les détails relatifs à la tâche de fenêtre de maintenance que vous avez créée.

   Linux

   ```
   aws ssm describe-maintenance-window-tasks \
       --window-id mw-0c50858d01EXAMPLE
   ```

Windows

```
aws ssm describe-maintenance-window-tasks ^
    --window-id mw-0c50858d01EXAMPLE
```

3. Le système renvoie des informations similaires à ce qui suit :

```
{
    "Tasks": [
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "WindowTaskId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE",
            "TaskArn": "AWS-RunShellScript",
            "Type": "RUN_COMMAND",
            "Targets": [
                {
                    "Key": "InstanceIds",
                    "Values": [
                        "i-02573cafcfEXAMPLE"
                    ]
                }
            ],
            "TaskParameters": {},
            "Priority": 10,
            "ServiceRoleArn": "arn:aws:iam::123456789012:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
            "MaxConcurrency": "1",
            "MaxErrors": "1"
        }
    ]
}
```

4. Attendez que la tâche ait eu le temps d'être exécutée, conformément à la planification que vous avez spécifiée dans Étape 1 : Créer la fenêtre de maintenance (AWS CLI) (p. 717). Par exemple, si vous avez spécifié **`--schedule "rate(5 minutes)"`**, attendez cinq minutes. Ensuite, exécutez la commande suivante pour afficher des informations sur toutes les exécutions qui se sont produites pour cette tâche.

Linux

```
aws ssm describe-maintenance-window-executions \
    --window-id mw-0c50858d01EXAMPLE
```

Windows

```
aws ssm describe-maintenance-window-executions ^
    --window-id mw-0c50858d01EXAMPLE
```

Le système renvoie des informations similaires à ce qui suit :

```
{
    "WindowExecutions": [
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "WindowExecutionId": "14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE",
            "Status": "SUCCESS",
            "StartTime": 1557593493.096,
            "EndTime": 1557593498.611
        }
```

```
    ]
}
```

Tip

Une fois que la tâche s'est terminée avec succès, vous pouvez diminuer la fréquence à laquelle la fenêtre de maintenance s'exécute. Par exemple, exécutez la commande suivante pour réduire la fréquence à une fois par semaine :

Linux

```
aws ssm update-maintenance-window \
    --window-id mw-0c50858d01EXAMPLE \
    --schedule "rate(7 days)"
```

Windows

```
aws ssm update-maintenance-window ^
    --window-id mw-0c50858d01EXAMPLE ^
    --schedule "rate(7 days)"
```

Pour de plus amples informations sur la planification des fenêtres de maintenance, veuillez consulter Référence : Expressions cron et rate pour Systems Manager (p. 1392) et Référence : Options de planification de la fenêtre de maintenance et de période active (p. 772).
Pour plus d'informations sur l'utilisation de l'AWS CLI pour modifier une fenêtre de maintenance, consultez Didacticiel : Mettre à jour une fenêtre de maintenance (AWS CLI) (p. 754).

Pour mettre en pratique l'exécution de commandes de l'AWS CLI pour afficher plus de détails sur votre tâche de fenêtre de maintenance et ses exécutions, passez à Didacticiel : Afficher les informations sur les tâches et les exécutions de tâches (AWS CLI) (p. 751).

À propos de la sortie de commande du didacticiel

L'utilisation de l'AWS CLI pour afficher la sortie de la commande Fonctionnalité Exécuter la commande associée avec vos exécutions de tâches de fenêtres de maintenance n'entre pas dans le cadre de ce didacticiel.

Toutefois, vous pouvez afficher ces données à l'aide de l'AWS CLI. (Vous pouvez également afficher la sortie dans la Systems Manager console ou dans un fichier journal stocké dans un compartiment S3, si vous avez configuré la fenêtre de maintenance pour y stocker la sortie de commande.) Vous constaterez que la sortie de la df commande sur une instance EC2 pour Linux est similaire à ce qui suit :

```
Filesystem 1K-blocks Used Available Use% Mounted on

devtmpfs 485716 0 485716 0% /dev

tmpfs 503624 0 503624 0% /dev/shm

tmpfs 503624 328 503296 1% /run

tmpfs 503624 0 503624 0% /sys/fs/cgroup

/dev/xvda1 8376300 1464160 6912140 18% /
```

La sortie de la commande ipconfig sur une instance EC2 pour Windows Server est similaire à ce qui suit :

```
Windows IP Configuration
```

```
Ethernet adapter Ethernet 2:

   Connection-specific DNS Suffix  . : example.com
   IPv4 Address. . . . . . . . . . . : 10.24.34.0/23
   Subnet Mask . . . . . . . . . . . : 255.255.255.255
   Default Gateway . . . . . . . . . : 0.0.0.0

Ethernet adapter Ethernet:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . : abc1.wa.example.net

Wireless LAN adapter Local Area Connection* 1:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

Wireless LAN adapter Wi-Fi:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::100b:c234:66d6:d24f%4
   IPv4 Address. . . . . . . . . . . : 192.0.2.0
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.0.2.0

Ethernet adapter Bluetooth Network Connection:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :
```

Exemples : Enregistrer les tâches avec une fenêtre de maintenance

Vous pouvez enregistrer une tâche Systems Manager Fonctionnalité Exécuter la commande avec une fenêtre de maintenance à l'aide de l'AWS CLI, comme illustré dans Étape 3 : Enregistrer une tâche avec la fenêtre de maintenance (AWS CLI) (p. 723). Vous pouvez également enregistrer des tâches pour les flux de travail Automation Systems Manager, les fonctions AWS Lambda et les tâches AWS Step Functions, comme illustré ci-dessous.

Dans cette rubrique, nous fournissons des exemples d'utilisation de la commande d'interface de ligne de commande `register-task-with-maintenance-window` pour enregistrer chacun des quatre types de tâches pris en charge avec une fenêtre de maintenance. Les exemples sont uniquement fournis à titre de démonstration. Mais vous pouvez les modifier afin de créer des commandes réelles d'enregistrement des tâches.

Utilisation de l'option --cli-input-json

Afin de mieux gérer vos options de tâches, vous pouvez utiliser l'option de commande `--cli-input-json`, avec les valeurs d'options référencées dans un fichier JSON.

Pour utiliser l'exemple de contenu de fichier JSON que nous fournissons dans les exemples suivants, effectuez les opérations suivantes sur votre machine locale :

1. Créez un fichier portant un nom comme `MyRunCommandTask.json`, `MyAutomationTask.json` ou tout autre nom de votre choix.
2. Copiez le contenu de notre exemple JSON dans le fichier.
3. Modifiez le contenu du fichier pour l'enregistrement de vos tâches, puis enregistrez le fichier.
4. Dans le répertoire où vous avez stocké le fichier, exécutez la commande suivante. Remplacez votre nom de fichier par *`MyFile.json`*.

Linux

```
aws ssm register-task-with-maintenance-window \
    --cli-input-json file://MyFile.json
```

Windows

```
aws ssm register-task-with-maintenance-window ^
    --cli-input-json file://MyFile.json
```

À propos des pseudo-paramètres

Dans certains exemples, nous utilisons des pseudo-paramètres comme méthode pour transmettre des informations d'ID à vos tâches. Par exemple, `{{RESOURCE_ID}}` est utilisé pour transmettre des ressources AWS à IDs l'automatisation, et aux Lambda tâches .Step Functions Pour plus d'informations sur les pseudo-paramètres dans les contenus `--task-invocation-parameters`, consultez À propos des pseudo-paramètres (p. 736).

En savoir plus

Pour plus d'informations sur certaines options `register-task-with-maintenance-window` fondamentales, consultez À propos des options register-task-with-maintenance-windows (p. 733).

Pour obtenir des informations complètes sur les options de commandes, consultez les rubriques suivantes :

- register-task-with-maintenance-window dans le AWS CLI Command Reference.
- RegisterTaskWithMaintenanceWindow dans le Référence d'API AWS Systems Manager.

Exemples d'enregistrement de tâches

Les sections suivantes fournissent un exemple de commande d'AWS CLI pour l'enregistrement d'un type de tâche pris en charge et un exemple de JSON qui peut être utilisé avec l'option `--cli-input-json`.

Enregistrer une tâche Systems Manager Fonctionnalité Exécuter la commande

Les exemples suivants montrent comment enregistrer des tâches Systems Manager Fonctionnalité Exécuter la commande avec une fenêtre de maintenance à l'aide de l'AWS CLI

AWS CLI Commande de l' :

Linux

```
aws ssm register-task-with-maintenance-window \
    --window-id mw-0c50858d01EXAMPLE \
    --task-arn "AWS-RunShellScript" \
    --max-concurrency 1 --max-errors 1 --priority 10 \
    --targets "Key=InstanceIds,Values=i-02573cafcfEXAMPLE" \
    --task-type "RUN_COMMAND" \
    --task-invocation-parameters "{"RunCommand":{"Parameters":{"commands":["df"]}}}"
```

Windows

```
aws ssm register-task-with-maintenance-window ^
    --window-id mw-0c50858d01EXAMPLE ^
    --task-arn "AWS-RunShellScript" ^
    --max-concurrency 1 --max-errors 1 --priority 10 ^
```

```
    --targets "Key=InstanceIds,Values=i-02573cafcfEXAMPLE" ^
    --task-type "RUN_COMMAND" ^
    --task-invocation-parameters "{"RunCommand":{"Parameters":{"commands":["df"]}}}"
```

Contenu JSON à utiliser avec **`--cli-input-json`** l'option de fichier :

```
{
    "TaskType": "RUN_COMMAND",
    "WindowId": "mw-0c50858d01EXAMPLE",
    "Description": "My Run Command task to update SSM Agent on an instance",
    "MaxConcurrency": "1",
    "MaxErrors": "1",
    "Name": "My-Run-Command-Task",
    "Priority": 10,
    "Targets": [
        {
            "Key": "WindowTargetIds",
            "Values": [
                "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
            ]
        }
    ],
    "TaskArn": "AWS-UpdateSSMAgent",
    "TaskInvocationParameters": {
        "RunCommand": {
            "Comment": "A TaskInvocationParameters test comment",
            "NotificationConfig": {
                "NotificationArn": "arn:aws:sns:us-east-2:123456789012:my-sns-topic-name",
                "NotificationEvents": [
                    "All"
                ],
                "NotificationType": "Invocation"
            },
            "OutputS3BucketName": "DOC-EXAMPLE-BUCKET",
            "OutputS3KeyPrefix": "DOC-EXAMPLE-FOLDER",
            "TimeoutSeconds": 3600
        }
    }
}
```

Enregistrement d'une tâche Systems Manager Automation

Les exemples suivants montrent comment enregistrer des tâches Systems Manager Automation avec une fenêtre de maintenance à l'aide de l'AWS CLI :

AWS CLI Commande de l' :

Linux

La commande suivante redémarre les instances EC2 appartenant au groupe cible de la fenêtre de maintenance avec l'ID e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE.

```
aws ssm register-task-with-maintenance-window \
    --window-id "mw-0c50858d01EXAMPLE" \
    --targets Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE \
    --task-arn "AWS-RestartEC2Instance" \
    --service-role-arn arn:aws:iam::123456789012:role/MyMaintenanceWindowServiceRole \
    --task-type AUTOMATION \
    --task-invocation-parameters
 "Automation={DocumentVersion=5,Parameters={instanceId='{{RESOURCE_ID}}'}}" \
    --priority 0 --max-concurrency 10 --max-errors 5 --name "My-Restart-EC2-Instances-
Automation-Task" \
```

```
    --description "Automation task to restart EC2 instances"
```

Windows

```
aws ssm register-task-with-maintenance-window ^
    --window-id "mw-0c50858d01EXAMPLE" ^
    --targets Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE ^
    --task-arn "AWS-RestartEC2Instance" ^
    --service-role-arn arn:aws:iam::123456789012:role/MyMaintenanceWindowServiceRole ^
    --task-type AUTOMATION ^
    --task-invocation-parameters
 "Automation={DocumentVersion=5,Parameters={instanceId='{{TARGET_ID}}'}}" ^
    --priority 0 --max-concurrency 10 --max-errors 5 --name "My-Restart-EC2-Instances-
Automation-Task" ^
    --description "Automation task to restart EC2 instances"
```

Contenu JSON à utiliser avec **`--cli-input-json`** l'option de fichier :

```
{
    "WindowId": "mw-0c50858d01EXAMPLE",
    "Targets": [
        {
            "Key": "WindowTargetIds",
            "Values": [
                "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
            ]
        }
    ],
    "TaskArn": "AWS-PatchInstanceWithRollback",
    "TaskType": "AUTOMATION",
    "MaxConcurrency": "10",
    "MaxErrors": "10",
    "TaskInvocationParameters": {
        "Automation": {
            "DocumentVersion": "1",
            "Parameters": {
                "instanceId": [
                    "{{RESOURCE_ID}}"
                ]
            }
        }
    }
}
```

Enregistrer une tâche AWS Lambda

Les exemples suivants montrent comment enregistrer des tâches de fonctions AWS Lambda avec une fenêtre de maintenance à l'aide de l'AWS CLI.

Pour ces exemples, l'utilisateur qui a créé la fonction Lambda l'a nommée `SSMrestart-my-instances` et a créé deux paramètres appelés `instanceId` et `targetType`.

> **Important**
>
> La IAM stratégie pour Fenêtres de maintenance nécessite que vous donniez un préfixe aux noms (ou alias) de Lambda fonction avec `SSM` . Avant de procéder à l'enregistrement de ce type de tâche, vous devez mettre à jour son nom dans AWS Lambda pour inclure SSM. Par exemple, si votre nom de fonction Lambda est `MyLambdaFunction`, remplacez-le par `SSMMyLambdaFunction`.

AWS CLI Commande de l' :

Linux

```
aws ssm register-task-with-maintenance-window \
    --window-id "mw-0c50858d01EXAMPLE" \
    --targets "Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE" \
    --priority 2 --max-concurrency 10 --max-errors 5 --name "My-Lambda-Example" \
    --description "A description for my LAMBDA example task" --task-type "LAMBDA" \
    --task-arn "arn:aws:lambda:us-east-2:123456789012:function:serverlessrepo-
SSMrestart-my-instances-C4JF9EXAMPLE" \
    --task-invocation-parameters '{"Lambda":{\"Payload\":{\"instanceId\":
\"{{RESOURCE_ID}}\",\"targetType\":\"{{TARGET_TYPE}}\"},"Qualifier": "$LATEST"}}'
```

Windows

```
aws ssm register-task-with-maintenance-window ^
    --window-id "mw-0c50858d01EXAMPLE" ^
    --targets "Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE" ^
    --priority 2 --max-concurrency 10 --max-errors 5 --name "My-Lambda-Example" ^
    --description "A description for my LAMBDA example task" --task-type "LAMBDA" ^
    --task-arn "arn:aws:lambda:us-east-2:123456789012:function:serverlessrepo-
SSMrestart-my-instances-C4JF9EXAMPLE" ^
    --task-invocation-parameters '{"Lambda":{\"Payload\":{\"instanceId\":
\"{{RESOURCE_ID}}\",\"targetType\":\"{{TARGET_TYPE}}\"},"Qualifier": "$LATEST"}}'
```

Contenu JSON à utiliser avec l'option **`--cli-input-json`** de fichier :

```
{
    "WindowId": "mw-0c50858d01EXAMPLE",
    "Targets": [
        {
            "Key": "WindowTargetIds",
            "Values": [
                "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
            ]
        }
    ],
    "TaskArn": "SSM_RestartMyInstances",
    "TaskType": "LAMBDA",
    "MaxConcurrency": "10",
    "MaxErrors": "10",
    "TaskInvocationParameters": {
        "Lambda": {
            "ClientContext": "ew0KICAi--truncated--0KIEXAMPLE",
            "Payload": "{ \"instanceId\": \"{{RESOURCE_ID}}\", \"targetType\":
 \"{{TARGET_TYPE}}\" }",
            "Qualifier": "$LATEST"
        }
    },
    "Name": "My-Lambda-Task",
    "Description": "A description for my LAMBDA task",
    "Priority": 5
}
```

Enregistrer une tâche AWS Step Functions

Les exemples suivants montrent comment enregistrer des tâches de machine d'état AWS Step Functions avec une fenêtre de maintenance à l'aide de l'AWS CLI.

Pour ces exemples, l'utilisateur qui a créé la machine d'état Step Functions à créé une machine d'état nommée `SSMMyStateMachine` avec un paramètre appelé `instanceId`.

> **Important**
>
> La IAM stratégie pour Fenêtres de maintenance exige que vous donniez un préfixe aux noms de machine dStep Functions'état avec `SSM` . Avant de procéder à l'enregistrement de ce type de tâche, vous devez mettre à jour son nom dans AWS Step Functions pour inclure `SSM` . Par exemple, si le nom de votre machine d'état est `MyStateMachine` , remplacez-le par `SSMMyStateMachine` .

AWS CLI Commande de l' :

Linux

```
aws ssm register-task-with-maintenance-window \
    --window-id "mw-0c50858d01EXAMPLE" \
    --targets "Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE" \
    --task-arn arn:aws:states:us-east-2:123456789012:stateMachine:SSMMyStateMachine-
MggiqEXAMPLE \
    --task-type STEP_FUNCTIONS \
    --task-invocation-parameters '{"StepFunctions":{"Input":"{\"instanceId\":
\"{{RESOURCE_ID}}\"}"}, "Name": "{{INVOCATION_ID}}"}' \
    --priority 0 --max-concurrency 10 --max-errors 5 \
    --name "My-Step-Functions-Task" --description "A description for my Step Functions
 task"
```

Windows

```
aws ssm register-task-with-maintenance-window ^
    --window-id "mw-0c50858d01EXAMPLE" ^
    --targets "Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE" ^
    --task-arn arn:aws:states:us-east-2:123456789012:stateMachine:SSMMyStateMachine-
MggiqEXAMPLE ^
    --task-type STEP_FUNCTIONS ^
    --task-invocation-parameters '{"StepFunctions":{"Input":"{\"instanceId\":
\"{{RESOURCE_ID}}\"}"}, "Name": "{{INVOCATION_ID}}"}' ^
    --priority 0 --max-concurrency 10 --max-errors 5 ^
    --name "My-Step-Functions-Task" --description "A description for my Step Functions
 task"
```

Contenu JSON à utiliser avec l'option **`--cli-input-json`** de fichier :

```
{
    "WindowId": "mw-0c50858d01EXAMPLE",
    "Targets": [
        {
            "Key": "WindowTargetIds",
            "Values": [
                "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
            ]
        }
    ],
    "TaskArn": "SSM_MyStateMachine",
    "TaskType": "STEP_FUNCTIONS",
    "MaxConcurrency": "10",
    "MaxErrors": "10",
    "TaskInvocationParameters": {
        "StepFunctions": {
            "Input": "{ \"instanceId\": \"{{TARGET_ID}}\" }",
            "Name": "{{INVOCATION_ID}}"
        }
    },
```

```
    "Name": "My-Step-Functions-Task",
    "Description": "A description for my Step Functions task",
    "Priority": 5
}
```

À propos des options register-task-with-maintenance-windows

La commande register-task-with-maintenance-window fournit plusieurs options pour configurer une tâche en fonction de vos besoins. Certaines sont requises, certaines sont facultatives et d'autres s'appliquent à un seul type de tâche de fenêtre de maintenance.

Cette rubrique fournit des informations sur certaines de ces options pour vous aider à utiliser les exemples de cette section du didacticiel. Pour plus d'informations sur toutes les options de commandes, consultez register-task-with-maintenance-window dans le AWS CLI Command Reference.

A propos de l'option **`--task-arn`**

L'option `--task-arn` est utilisée pour spécifier la ressource utilisée par la tâche pendant l'exécution. La valeur que vous spécifiez dépend du type de tâche que vous enregistrez, comme décrit dans le tableau suivant.

TaskArnFormats pour les tâches de fenêtre de maintenance

Type de tâche de fenêtre de maintenance	TaskArnvaleur
`RUN_COMMAND`et**`AUTOMATION`**	`TaskArn` est le nom ou l'ARN du document SSM. Exemple : `AWS-RunBatchShellScript` -ou- `arn:aws:ssm:us-east-2:111122223333:document/My-Document`.
`LAMBDA`	`TaskArn` est le nom de fonction ou l'ARN. Exemple : `SSMMy-Lambda-Function` -ou- `arn:aws:lambda:us-east-2:111122223333:function:SSMMyLambdaFunction`. Important La IAM stratégie pour les fenêtres de maintenance nécessite que vous donniez un préfixe aux noms de Lambda fonction (ou alias) avec `SSM` . Avant d'enregistrer ce type de tâche, vous devez mettre à jour son nom dans AWS Lambda pour inclure `SSM` . Par exemple, si le nom de votre Lambda fonction est `MyLambdaFunction` , remplacez-le par `SSMMyLambdaFunction` .
`STEP_FUNCTIONS`	`TaskArn` est l'ARN de la machine d'état. Exemple :

Type de tâche de fenêtre de maintenance	TaskArnvaleur
	`arn:aws:states:us-east-2:111122223333:stateMachine:SSMMyStateMachin` **Important** La IAM stratégie pour les fenêtres de maintenance exige que vous donniez un préfixe aux noms de machines dStep Functions'état avec `SSM` . Avant d'enregistrer ce type de tâche, vous devez mettre à jour son nom dans AWS Step Functions pour inclure `SSM` . Par exemple, si le nom de votre machine d'état est `MyStateMachine` , remplacez-le par `SSMMyStateMachine` .

A propos de l'option **`--service-role-arn`**

Rôle pour Systems Manager à assumer lors de l'exécution de la tâche de fenêtre de maintenance.

La spécification d'un ARN de rôle de service est facultative. Si vous ne spécifiez aucun ARN de rôle de service, Systems Manager crée un rôle lié à un service ou utilise le rôle lié à un service présent dans votre compte.

Notez que le rôle lié au service Systems Manager ne fournit pas les autorisations nécessaires pour tous les scénarios. Pour plus d'informations, veuillez consulter Dois-je utiliser un rôle lié à un service ou un rôle de service personnalisé pour exécuter des tâches de fenêtre de maintenance ? (p. 691)

A propos de l'option **`--task-invocation-parameters`**

L'option `--task-invocation-parameters` permet de spécifier les paramètres qui sont spécifiques à chacun des quatre types de tâches. Les paramètres pris en charge pour chacun des quatre types de tâches sont décrits dans le tableau suivant.

> **Note**
>
> Pour plus d'informations sur l'utilisation des pseudo-paramètres dans le contenu `--task-invocation-parameters`, comme {{TARGET_ID}}, consultez À propos des pseudo-paramètres (p. 736).

Options de paramètres des invocations de tâches pour les tâches de fenêtres de maintenance

Type de tâche de fenêtre de maintenance	Paramètres disponibles	Exemple
RUN_COMMAND	Commentaire DocumentHash DocumentHashType NotificationConfig OutputS3BucketName OutPutS3KeyPrefix Parameters	`"TaskInvocationParameters": { "RunCommand": { "Comment": "My Run Command task comment", "DocumentHash": "6554ed3d--truncated--5EXAMPLE", "DocumentHashType": "Sha256", "NotificationConfig": {`

Type de tâche de fenêtre de maintenance	Paramètres disponibles	Exemple

	ServiceRoleArn
	
	TimeoutSeconds
		`"NotificationArn": "arn:aws:sns:us-east-2:123456789012:my-sns-topic-name",`
		
		`"NotificationEvents": [`
		
		`"FAILURE" ],`
		
		`"NotificationType": "Invocation" },`
		
		`"OutputS3BucketName": "DOC-EXAMPLE-BUCKET",`
		
		`"OutputS3KeyPrefix": "DOC-EXAMPLE-FOLDER", "Parameters": { "commands": [ "Get-ChildItem$env: temp-Recurse| Remove-Item-Recurse-force" ] },`
		
		`"ServiceRoleArn": "arn:aws:iam::123456789012:role/MyMaintenanceWindowServiceRole",`
		
		`"TimeoutSeconds": 3600 } }`

AUTOMATISATION	DocumentVersion
	
	Parameters
		`"TaskInvocationParameters": { "Automation": {`
		
		`"DocumentVersion": "3", "Parameters": {`
		
		`"instanceid": [`
		
		`"{{TARGET_ID}}" ] } } }`

Type de tâche de fenêtre de maintenance	Paramètres disponibles	Exemple
LAMBDA	ClientContext Charge utile Qualificateur	`"TaskInvocationParameters": { "Lambda": { "ClientContext": "ew0KICAi--truncated--0KIEXAMPLE", "Payload": "{ \"targetId\": \"{{TARGET_ID}}\", \"targetType\": \"{{TARGET_TYPE}}\" }", "Qualifier": "$LATEST" } }`
STEP_FUNCTIONS	Entrées Nom	`"TaskInvocationParameters": { "StepFunctions": { "Input": "{ \"targetId\": \"{{TARGET_ID}}\" }", "Name": "{{INVOCATION_ID}}" } }`

À propos des pseudo-paramètres

Lorsque vous enregistrez une tâche, vous utilisez l'option `--task-invocation-parameters` pour spécifier les paramètres qui sont spécifiques à chacun des quatre types de tâches. Vous pouvez également référencer certaines valeurs à l'aide de la syntaxe des pseudo-paramètres, comme `{{RESOURCE_ID}}` , et `{{TARGET_TYPE}}` .`{{WINDOW_TARGET_ID}}` Quand la tâche de fenêtre de maintenance s'exécute, elle transmet les valeurs correctes au lieu des espaces réservés des pseudo-paramètres. La liste complète des pseudo-paramètres que vous pouvez utiliser est fournie dans Pseudo-paramètres pris en charge (p. 738).

> **Important**
>
> Pour le type de cible `RESOURCE_GROUP` , selon le format d'ID nécessaire pour la tâche, vous pouvez choisir d'utiliser `{{TARGET_ID}}` et `{{RESOURCE_ID}}` de référencer la ressource lorsque votre tâche est exécutée. `{{TARGET_ID}}`renvoie l'ARN complet de la ressource. `{{RESOURCE_ID}}`renvoie uniquement un nom ou un ID plus court de la ressource, comme illustré dans ces exemples.
>
> - `{{TARGET_ID}}`format :`arn:aws:ec2:us-east-1:123456789012:instance/i-02573cafcfEXAMPLE`
> - `{{RESOURCE_ID}}`format :`i-02573cafcfEXAMPLE`
>
> Pour le type de cible `INSTANCE`, les paramètres `{{TARGET_ID}}` et `{{RESOURCE_ID}}` ne donnent que l'ID d'instance. Pour plus d'informations, consultez la section Pseudo-paramètres pris en charge (p. 738).

Exemples de pseudo-paramètres

Supposons que votre charge utile pour une tâche Lambda doive référencer une instance par son ID.

Que vous utilisiez un `INSTANCE` ou une cible de fenêtre de maintenance `RESOURCE_GROUP`, cela peut être réalisé en utilisant le pseudo-paramètre `{{RESOURCE_ID}}` :

```
"TaskArn": "arn:aws:lambda:us-east-2:111122223333:function:SSMTestFunction",
    "TaskType": "LAMBDA",
    "TaskInvocationParameters": {
        "Lambda": {
            "ClientContext": "ew0KICAi--truncated--0KIEXAMPLE",
            "Payload": "{ \"instanceId\": \"{{RESOURCE_ID}}\" }",
            "Qualifier": "$LATEST"
        }
    }
```

Si votre tâche Lambda est destinée à s'exécuter sur un autre type de cible pris en charge en plus des instances EC2, par exemple une table DynamoDB, la même syntaxe peut être utilisée et `{{RESOURCE_ID}}` donne le nom de la table uniquement. Toutefois, si vous avez besoin de l'ARN complet de la table, utilisez `{{TARGET_ID}}`, comme indiqué dans l'exemple suivant.

```
"TaskArn": "arn:aws:lambda:us-east-2:111122223333:function:SSMTestFunction",
    "TaskType": "LAMBDA",
    "TaskInvocationParameters": {
        "Lambda": {
            "ClientContext": "ew0KICAi--truncated--0KIEXAMPLE",
            "Payload": "{ \"tableArn\": \"{{TARGET_ID}}\" }",
            "Qualifier": "$LATEST"
        }
    }
```

La même syntaxe fonctionne pour les instances de ciblage ou d'autres types de ressources. Lorsque plusieurs types de ressources ont été ajoutés à un groupe de ressources, la tâche s'exécute sur chacune des ressources appropriées.

> **Important**
>
> Tous les types de ressources qui peuvent être inclus dans un groupe de ressources ne donnent pas de valeur pour le paramètre `{{RESOURCE_ID}}`. Pour obtenir la liste des types de ressources pris en charge, consultez Pseudo-paramètres pris en charge (p. 738) .

Autre exemple, pour exécuter une tâche Automation qui arrête vos instances EC2, vous spécifiez le document SSM `AWS-StopEC2Instance` comme valeur `TaskArn` et vous utilisez le pseudo-paramètre `{{RESOURCE_ID}}` :

```
"TaskArn": "AWS-StopEC2Instance",
    "TaskType": "AUTOMATION"
    "TaskInvocationParameters": {
        "Automation": {
            "DocumentVersion": "1",
            "Parameters": {
                "instanceId": [
                    "{{RESOURCE_ID}}"
                ]
            }
        }
    }
```

Pour exécuter une tâche d'automatisation qui copie un instantané d'un volume Amazon Elastic Block Store (Amazon EBS), spécifiez le document `AWS-CopySnapshot` SSM comme valeur `TaskArn` et utilisez le pseudo-paramètre `{{RESOURCE_ID}}` :

```
"TaskArn": "AWS-CopySnapshot",
    "TaskType": "AUTOMATION"
    "TaskInvocationParameters": {
        "Automation": {
            "DocumentVersion": "1",
            "Parameters": {
                "SourceRegion": "us-east-2",
                "targetType":"RESOURCE_GROUP",
                "SnapshotId": [
                    "{{RESOURCE_ID}}"
                ]
            }
        }
    }
```

Pseudo-paramètres pris en charge

La liste suivante décrit les pseudo-paramètres que vous pouvez spécifier à l'aide de la syntaxe `{{PSEUDO_PARAMETER}}` dans l'option `--task-invocation-parameters`.

- **`WINDOW_ID`**: ID de la fenêtre de maintenance cible.
- **`WINDOW_TASK_ID`**: ID de la tâche de fenêtre en cours d'exécution.
- **`WINDOW_TARGET_ID`**: ID de la cible de fenêtre qui inclut la cible (ID cible).
- **`WINDOW_EXECUTION_ID`**: ID de l'exécution de la fenêtre actuelle.
- **`TASK_EXECUTION_ID`**: ID de l'exécution de tâche actuelle.
- **`INVOCATION_ID`**: ID de l'appel actuel.
- **`TARGET_TYPE`**: Type de cible. Les types pris en charge incluent `RESOURCE_GROUP` et `INSTANCE`.
- **`TARGET_ID`**:

  Si le type de cible que vous spécifiez est `INSTANCE`, le pseudo-paramètre `TARGET_ID` est remplacé par l'ID de l'instance ; par exemple `i-078a280217EXAMPLE`.

  Si le type de cible que vous spécifiez est `RESOURCE_GROUP` , la valeur référencée pour l'exécution de la tâche est l'ARN complet de la ressource ; par exemple : `arn:aws:ec2:us-east-1:123456789012:instance/i-078a280217EXAMPLE`. Le tableau suivant fournit des exemples de `TARGET_ID` valeurs pour des types de ressources spécifiques d'un groupe de ressources.

Type de ressource	Exemple TARGET_ID	
AWS::EC2::Instance	`arn:aws:ec2:us-east-1:123456789012:instance/i-078a280217EXAMPLE`	
AWS::EC2::Image	`arn:aws:ec2:us-east-1:123456789012:image/ami-02250b3732EXAMPLE`	
AWS::EC2::SecurityGroup	`arn:aws:ec2:us-east-1:123456789012:security-group/sg-cEXAMPLE`	

Type de ressource	Exemple TARGET_ID	
AWS::EC2::Snapshot	`arn:aws:ec2:us-east-1:123456789012:snapshot/snap-03866bf003EXAMPLE`	
AWS::EC2::Volume	`arn:aws:ec2:us-east-1:123456789012:volume/vol-0912e04d78EXAMPLE`	
AWS::DynamoDB::Table	`arn:aws:dynamodb:us-east-1:123456789012:table/MyTable`	
AWS::RDS::DBInstance	`arn:aws:rds:us-east-1:123456789012:db:My-SQL-Instance`	
AWS::S3::Bucket	`arn:aws:s3:::DOC-EXAMPLE-BUCKET`	
AWS::SSM::ManagedInstance	`arn:aws:ssm:us-east-1:123456789012:managed-instance/mi-0feadcf2d9EXAMPLE`	

- **`RESOURCE_ID`**: ID court d'un type de ressource contenu dans un groupe de ressources. Le tableau suivant fournit des exemples de valeurs `RESOURCE_ID` pour des types de ressources particuliers d'un groupe de ressources.

Type de ressource	Exemple RESOURCE_ID	
AWS::EC2::Instance	`i-078a280217EXAMPLE`	
AWS::EC2::Image	`ami-02250b3732EXAMPLE`	
AWS::EC2::SecurityGroup	`sg-cEXAMPLE`	
AWS::EC2::Snapshot	`snap-03866bf003EXAMPLE`	
AWS::EC2::Volume	`vol-0912e04d78EXAMPLE`	
AWS::DynamoDB::Table	`MyTable`	
AWS::RDS::DBInstance	`My-SQL-Instance`	
AWS::S3::Bucket	`DOC-EXAMPLE-BUCKET`	
AWS::SSM::ManagedInstance	`mi-0feadcf2d9EXAMPLE`	

> **Note**
>
> Si le groupe de ressources AWS que vous spécifiez inclut des types de ressources qui ne donnent pas de valeur `RESOURCE_ID` et ne sont pas répertoriés dans le tableau ci-dessus, le paramètre `RESOURCE_ID` n'est pas renseigné. Une invocation d'exécution se produira toujours pour cette ressource. Dans ces cas, utilisez plutôt le pseudo-paramètre `TARGET_ID`, qui sera remplacé par l'ARN complet de la ressource.

Didacticiel : Affichage des informations sur les fenêtres de maintenance (AWS CLI)

Ce didacticiel inclut des commandes pour vous aider à mettre à jour ou obtenir les informations sur vos fenêtres de maintenance, tâches, exécutions et appels. Les exemples sont organisés par commande pour montrer comment utiliser les options de commande afin de filtrer le type de détail que vous souhaitez afficher.

Tout au long des étapes de ce didacticiel, remplacez les valeurs indiquées en italiques et en *rouge* par vos propres options et identifiants. Par exemple, remplacez l'ID de la fenêtre de maintenance *mw-0c50858d01EXAMPLE* et l'ID d' instance *i-02573cafcfEXAMPLE* par des ID de ressource que vous créez.

Pour plus d'informations sur l'installation et la configuration de l'interface de ligne de commande, consultez Installation de l'AWS Command Line Interface et Configuration de l'AWS Command Line Interface.

Exemples de commandes

- Exemples pour « describe-maintenance-windows » (p. 740)
- Exemples pour « describe-maintenance-window-targets » (p. 742)
- Exemples pour « describe-maintenance-window-tasks » (p. 743)
- Exemples pour « describe-maintenance-windows-for-target » (p. 746)
- Exemples pour « describe-maintenance-window-executions » (p. 746)
- Exemples pour « describe-maintenance-window-schedule » (p. 748)

Exemples pour « describe-maintenance-windows »

Affichage de toutes les fenêtres de maintenance de votre compte AWS

Exécutez la commande suivante :

```
aws ssm describe-maintenance-windows
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
   "WindowIdentities":[
      {
         "WindowId":"mw-0c50858d01EXAMPLE",
         "Name":"My-First-Maintenance-Window",
         "Enabled":true,
         "Duration":2,
         "Cutoff":0,
         "NextExecutionTime": "2019-05-18T17:01:01.137Z"
      },
      {
         "WindowId":"mw-9a8b7c6d5eEXAMPLE",
         "Name":"My-Second-Maintenance-Window",
         "Enabled":true,
         "Duration":4,
         "Cutoff":1,
         "NextExecutionTime": "2019-05-30T03:30:00.137Z"
      },
   ]
}
```

Affichage de toutes les fenêtres de maintenance activées

Exécutez la commande suivante :

```
aws ssm describe-maintenance-windows --filters "Key=Enabled,Values=true"
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
   "WindowIdentities":[
      {
         "WindowId":"mw-0c50858d01EXAMPLE",
         "Name":"My-First-Maintenance-Window",
         "Enabled":true,
         "Duration":2,
         "Cutoff":0,
         "NextExecutionTime": "2019-05-18T17:01:01.137Z"
      },
      {
         "WindowId":"mw-9a8b7c6d5eEXAMPLE",
         "Name":"My-Second-Maintenance-Window",
         "Enabled":true,
         "Duration":4,
         "Cutoff":1,
         "NextExecutionTime": "2019-05-30T03:30:00.137Z"
      },
   ]
}
```

Affichage de toutes les fenêtres de maintenance désactivées

Exécutez la commande suivante :

```
aws ssm describe-maintenance-windows --filters "Key=Enabled,Values=false"
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
    "WindowIdentities": [
        {
            "WindowId": "mw-6e5c9d4b7cEXAMPLE",
            "Name": "My-Disabled-Maintenance-Window",
            "Enabled": false,
            "Duration": 2,
            "Cutoff": 1
        }
    ]
}
```

Répertorier toutes les fenêtres de maintenance dont les noms commencent par un préfixe donné

Exécutez la commande suivante :

```
aws ssm describe-maintenance-windows --filters "Key=Name,Values=My"
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
    "WindowIdentities": [
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Name": "My-First-Maintenance-Window",
```

```
            "Enabled": true,
            "Duration": 2,
            "Cutoff": 0,
            "NextExecutionTime": "2019-05-18T17:01:01.137Z"
        },
        {
            "WindowId": "mw-9a8b7c6d5eEXAMPLE",
            "Name": "My-Second-Maintenance-Window",
            "Enabled": true,
            "Duration": 4,
            "Cutoff": 1,
            "NextExecutionTime": "2019-05-30T03:30:00.137Z"
        },
        {
            "WindowId": "mw-6e5c9d4b7cEXAMPLE",
            "Name": "My-Disabled-Maintenance-Window",
            "Enabled": false,
            "Duration": 2,
            "Cutoff": 1
        }
    ]
}
```

Exemples pour « describe-maintenance-window-targets »

Affichage des cibles d'une fenêtre de maintenance correspondant à une valeur d'Information d'un propriétaire spécifique

Exécutez la commande suivante :

Linux

```
aws ssm describe-maintenance-window-targets \
    --window-id "mw-6e5c9d4b7cEXAMPLE" \
    --filters "Key=OwnerInformation,Values=CostCenter1"
```

Windows

```
aws ssm describe-maintenance-window-targets ^
    --window-id "mw-6e5c9d4b7cEXAMPLE" ^
    --filters "Key=OwnerInformation,Values=CostCenter1"
```

> **Note**
>
> Les clés de filtres prises en charge sont `Type`, `WindowTargetId` et `OwnerInformation`.

Le système renvoie des informations similaires à celles qui suivent :

```
{
    "Targets": [
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "WindowTargetId": "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE",
            "ResourceType": "INSTANCE",
            "Targets": [
                {
                    "Key": "tag:Name",
                    "Values": [
                        "Production"
                    ]
                }
```

```
            ],
            "OwnerInformation": "CostCenter1",
            "Name": "Target1"
        }
    ]
}
```

Exemples pour « describe-maintenance-window-tasks »

Afficher toutes les tâches enregistrées qui appellent la fonctionnalité Fonctionnalité Exécuter la commande AWS-RunPowerShellScript

Exécutez la commande suivante :

Linux

```
aws ssm describe-maintenance-window-tasks \
    --window-id "mw-0c50858d01EXAMPLE" \
    --filters "Key=TaskArn,Values=AWS-RunPowerShellScript"
```

Windows

```
aws ssm describe-maintenance-window-tasks ^
    --window-id "mw-0c50858d01EXAMPLE" ^
    --filters "Key=TaskArn,Values=AWS-RunPowerShellScript"
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
   "Tasks":[
      {
         "ServiceRoleArn": "arn:aws:iam::111122223333:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
         "MaxErrors":"1",
         "TaskArn":"AWS-RunPowerShellScript",
         "MaxConcurrency":"1",
         "WindowTaskId":"4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE",
         "TaskParameters":{
            "commands":{
               "Values":[
                  "driverquery.exe"
               ]
            }
         },
         "Priority":3,
         "Type":"RUN_COMMAND",
         "Targets":[
            {
               "TaskTargetId":"i-02573cafcfEXAMPLE",
               "TaskTargetType":"INSTANCE"
            }
         ]
      },
      {
         "ServiceRoleArn":"arn:aws:iam::111122223333:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
         "MaxErrors":"1",
         "TaskArn":"AWS-RunPowerShellScript",
         "MaxConcurrency":"1",
         "WindowTaskId":"4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE",
         "TaskParameters":{
```

```
            "commands":{
               "Values":[
                  "ipconfig"
               ]
            }
         },
         "Priority":1,
         "Type":"RUN_COMMAND",
         "Targets":[
            {
               "TaskTargetId":"i-02573cafcfEXAMPLE",
               "TaskTargetType":"WINDOW_TARGET"
            }
         ]
      }
   ]
}
```

Afficher toutes les tâches enregistrées qui ont une priorité égale à « 3 »

Exécutez la commande suivante :

Linux

```
aws ssm describe-maintenance-window-tasks \
    --window-id "mw-9a8b7c6d5eEXAMPLE" \
    --filters "Key=Priority,Values=3"
```

Windows

```
aws ssm describe-maintenance-window-tasks ^
    --window-id "mw-9a8b7c6d5eEXAMPLE" ^
    --filters "Key=Priority,Values=3"
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
   "Tasks":[
      {
         "ServiceRoleArn":"arn:aws:iam::111122223333:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
         "MaxErrors":"1",
         "TaskArn":"AWS-RunPowerShellScript",
         "MaxConcurrency":"1",
         "WindowTaskId":"4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE",
         "TaskParameters":{
            "commands":{
               "Values":[
                  "driverquery.exe"
               ]
            }
         },
         "Priority":3,
         "Type":"RUN_COMMAND",
         "Targets":[
            {
               "TaskTargetId":"i-02573cafcfEXAMPLE",
               "TaskTargetType":"INSTANCE"
            }
         ]
      }
```

```
    ]
}
```

Afficher toutes les tâches enregistrées qui ont une priorité égale à « 1 » et qui utilisent la fonctionnalité Fonctionnalité Exécuter la commande

Exécutez la commande suivante :

Linux

```
aws ssm describe-maintenance-window-tasks \
    --window-id "mw-0c50858d01EXAMPLE" \
    --filters "Key=Priority,Values=1" "Key=TaskType,Values=RUN_COMMAND"
```

Windows

```
aws ssm describe-maintenance-window-tasks ^
    --window-id "mw-0c50858d01EXAMPLE" ^
    --filters "Key=Priority,Values=1" "Key=TaskType,Values=RUN_COMMAND"
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
    "Tasks": [
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "WindowTaskId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE",
            "TaskArn": "AWS-RunShellScript",
            "Type": "RUN_COMMAND",
            "Targets": [
                {
                    "Key": "InstanceIds",
                    "Values": [
                        "i-02573cafcfEXAMPLE"
                    ]
                }
            ],
            "TaskParameters": {},
            "Priority": 1,
            "ServiceRoleArn": "arn:aws:iam::111122223333:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
            "MaxConcurrency": "1",
            "MaxErrors": "1"
        },
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "WindowTaskId": "8a5c4629-31b0-4edd-8aea-33698EXAMPLE",
            "TaskArn": "AWS-UpdateSSMAgent",
            "Type": "RUN_COMMAND",
            "Targets": [
                {
                    "Key": "InstanceIds",
                    "Values": [
                        "i-0471e04240EXAMPLE"
                    ]
                }
            ],
            "TaskParameters": {},
            "Priority": 1,
            "ServiceRoleArn": "arn:aws:iam::111122223333:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM"",
```

```
            "MaxConcurrency": "1",
            "MaxErrors": "1",
            "Name": "My-Run-Command-Task",
            "Description": "My Run Command task to update SSM Agent on an instance"
        }
    ]
}
```

Exemples pour « describe-maintenance-windows-for-target »

Répertorier les informations sur les cibles ou tâches de fenêtre de maintenance associées à une instance spécifique

Exécutez la commande suivante :

Linux

```
aws ssm describe-maintenance-windows-for-target \
    --resource-type INSTANCE \
    --targets "Key=InstanceIds,Values=i-02573cafcfEXAMPLE" \
    --max-results 10
```

Windows

```
aws ssm describe-maintenance-windows-for-target ^
    --resource-type INSTANCE ^
    --targets "Key=InstanceIds,Values=i-02573cafcfEXAMPLE" ^
    --max-results 10
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
    "WindowIdentities": [
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Name": "My-First-Maintenance-Window"
        },
        {
            "WindowId": "mw-9a8b7c6d5eEXAMPLE",
            "Name": "My-Second-Maintenance-Window"
        }
    ]
}
```

Exemples pour « describe-maintenance-window-executions »

Répertorier toutes les tâches exécutées avant une date donnée

Exécutez la commande suivante :

Linux

```
aws ssm describe-maintenance-window-executions \
    --window-id "mw-9a8b7c6d5eEXAMPLE" \
    --filters "Key=ExecutedBefore,Values=2019-05-12T05:00:00Z"
```

Windows

```
aws ssm describe-maintenance-window-executions ^
```

```
    --window-id "mw-9a8b7c6d5eEXAMPLE" ^
    --filters "Key=ExecutedBefore,Values=2019-05-12T05:00:00Z"
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
    "WindowExecutions": [
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "WindowExecutionId": "14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE",
            "Status": "FAILED",
            "StatusDetails": "The following SSM parameters are invalid: LevelUp",
            "StartTime": 1557617747.993,
            "EndTime": 1557617748.101
        },
        {
            "WindowId": "mw-9a8b7c6d5eEXAMPLE",
            "WindowExecutionId": "791b72e0-f0da-4021-8b35-f95dfEXAMPLE",
            "Status": "SUCCESS",
            "StartTime": 1557594085.428,
            "EndTime": 1557594090.978
        },
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "WindowExecutionId": "ecec60fa-6bb0-4d26-98c7-140308EXAMPLE",
            "Status": "SUCCESS",
            "StartTime": 1557593793.483,
            "EndTime": 1557593798.978
        }
    ]
}
```

Répertorier toutes les tâches exécutées après une date donnée

Exécutez la commande suivante :

Linux

```
aws ssm describe-maintenance-window-executions \
    --window-id "mw-9a8b7c6d5eEXAMPLE" \
    --filters "Key=ExecutedAfter,Values=2018-12-31T17:00:00Z"
```

Windows

```
aws ssm describe-maintenance-window-executions ^
    --window-id "mw-9a8b7c6d5eEXAMPLE" ^
    --filters "Key=ExecutedAfter,Values=2018-12-31T17:00:00Z"
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
    "WindowExecutions": [
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "WindowExecutionId": "14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE",
            "Status": "FAILED",
            "StatusDetails": "The following SSM parameters are invalid: LevelUp",
            "StartTime": 1557617747.993,
            "EndTime": 1557617748.101
```

```
        },
        {
            "WindowId": "mw-9a8b7c6d5eEXAMPLE",
            "WindowExecutionId": "791b72e0-f0da-4021-8b35-f95dfEXAMPLE",
            "Status": "SUCCESS",
            "StartTime": 1557594085.428,
            "EndTime": 1557594090.978
        },
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "WindowExecutionId": "ecec60fa-6bb0-4d26-98c7-140308EXAMPLE",
            "Status": "SUCCESS",
            "StartTime": 1557593793.483,
            "EndTime": 1557593798.978
        }
    ]
}
```

Exemples pour « describe-maintenance-window-schedule »

Afficher les dix exécutions de fenêtre de maintenance planifiées suivantes pour une instance particulière

Exécutez la commande suivante :

Linux

```
aws ssm describe-maintenance-window-schedule \
    --resource-type INSTANCE \
    --targets "Key=InstanceIds,Values=i-07782c72faEXAMPLE" \
    --max-results 10
```

Windows

```
aws ssm describe-maintenance-window-schedule ^
    --resource-type INSTANCE ^
    --targets "Key=InstanceIds,Values=i-07782c72faEXAMPLE" ^
    --max-results 10
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
    "ScheduledWindowExecutions": [
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Name": "My-First-Maintenance-Window",
            "ExecutionTime": "2019-05-18T23:35:24.902Z"
        },
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Name": "My-First-Maintenance-Window",
            "ExecutionTime": "2019-05-25T23:35:24.902Z"
        },
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Name": "My-First-Maintenance-Window",
            "ExecutionTime": "2019-06-01T23:35:24.902Z"
        },
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Name": "My-First-Maintenance-Window",
            "ExecutionTime": "2019-06-08T23:35:24.902Z"
```

```
        },
        {
            "WindowId": "mw-9a8b7c6d5eEXAMPLE",
            "Name": "My-Second-Maintenance-Window",
            "ExecutionTime": "2019-06-15T23:35:24.902Z"
        },
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Name": "My-First-Maintenance-Window",
            "ExecutionTime": "2019-06-22T23:35:24.902Z"
        },
        {
            "WindowId": "mw-9a8b7c6d5eEXAMPLE",
            "Name": "My-Second-Maintenance-Window",
            "ExecutionTime": "2019-06-29T23:35:24.902Z"
        },
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Name": "My-First-Maintenance-Window",
            "ExecutionTime": "2019-07-06T23:35:24.902Z"
        },
        {
            "WindowId": "mw-9a8b7c6d5eEXAMPLE",
            "Name": "My-Second-Maintenance-Window",
            "ExecutionTime": "2019-07-13T23:35:24.902Z"
        },
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Name": "My-First-Maintenance-Window",
            "ExecutionTime": "2019-07-20T23:35:24.902Z"
        }
    ],
    "NextToken": "AAEABUXdceT92FvtKld/dGHELj5Mi+GKW/EXAMPLE"
}
```

Afficher la planification de fenêtre de maintenance pour les instances balisées avec une paire clé-valeur donnée

Exécutez la commande suivante :

Linux

```
aws ssm describe-maintenance-window-schedule \
    --resource-type INSTANCE \
    --targets "Key=tag:prod,Values=rhel7"
```

Windows

```
aws ssm describe-maintenance-window-schedule ^
    --resource-type INSTANCE ^
    --targets "Key=tag:prod,Values=rhel7"
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
    "ScheduledWindowExecutions": [
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Name": "DemoRateStartDate",
            "ExecutionTime": "2019-10-20T05:34:56-07:00"
        },
```

```
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Name": "DemoRateStartDate",
            "ExecutionTime": "2019-10-21T05:34:56-07:00"
        },
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Name": "DemoRateStartDate",
            "ExecutionTime": "2019-10-22T05:34:56-07:00"
        },
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Name": "DemoRateStartDate",
            "ExecutionTime": "2019-10-23T05:34:56-07:00"
        },
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Name": "DemoRateStartDate",
            "ExecutionTime": "2019-10-24T05:34:56-07:00"
        }
    ],
    "NextToken": "AAEABccwSXqQRGKiTZ1yzGELR6cxW4W/EXAMPLE"
}
```

Afficher les heures de début des quatre prochaines exécutions d'une fenêtre de maintenance

Exécutez la commande suivante :

Linux

```
aws ssm describe-maintenance-window-schedule \
    --window-id "mw-0c50858d01EXAMPLE" \
    --max-results "4"
```

Windows

```
aws ssm describe-maintenance-window-schedule ^
    --window-id "mw-0c50858d01EXAMPLE" ^
    --max-results "4"
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
    "WindowSchedule": [
        {
            "ScheduledWindowExecutions": [
                {
                    "ExecutionTime": "2019-10-04T10:10:10Z",
                    "Name": "My-First-Maintenance-Window",
                    "WindowId": "mw-0c50858d01EXAMPLE"
                },
                {
                    "ExecutionTime": "2019-10-11T10:10:10Z",
                    "Name": "My-First-Maintenance-Window",
                    "WindowId": "mw-0c50858d01EXAMPLE"
                },
                {
                    "ExecutionTime": "2019-10-18T10:10:10Z",
                    "Name": "My-First-Maintenance-Window",
                    "WindowId": "mw-0c50858d01EXAMPLE"
                },
                {
```

```
                    "ExecutionTime": "2019-10-25T10:10:10Z",
                    "Name": "My-First-Maintenance-Window",
                    "WindowId": "mw-0c50858d01EXAMPLE"
                }
            ]
        }
    ]
}
```

Didacticiel : Afficher les informations sur les tâches et les exécutions de tâches (AWS CLI)

Ce didacticiel montre comment utiliser l'AWS CLI pour afficher des informations détaillées sur les exécutions de tâches de fenêtres de maintenance terminées.

Si vous passez directement à ce didacticiel à partir de Didacticiel : Créer et configurer une fenêtre de maintenance (AWS CLI) (p. 717), assurez-vous d'avoir laissé assez de temps à votre fenêtre de maintenance pour qu'elle s'exécute au moins une fois afin de pouvoir afficher ses résultats d'exécution.

Tout au long des étapes de ce didacticiel, remplacez les valeurs indiquées en italiques et en *rouge* par vos propres options et identifiants. Par exemple, remplacez l'ID de la fenêtre de maintenance *mw-0c50858d01EXAMPLE* et l'ID d' instance *i-02573cafcfEXAMPLE* par des ID de ressource que vous créez.

Pour afficher des informations sur les tâches et les exécutions de tâches (AWS CLI)

1. Exécutez la commande suivante pour afficher la liste des exécutions de tâche pour une fenêtre de maintenance spécifique.

   Linux

   ```
   aws ssm describe-maintenance-window-executions \
       --window-id "mw-0c50858d01EXAMPLE"
   ```

   Windows

   ```
   aws ssm describe-maintenance-window-executions ^
       --window-id "mw-0c50858d01EXAMPLE"
   ```

   Le système renvoie des informations similaires à celles qui suivent.

   ```
   {
       "WindowExecutions": [
           {
               "WindowId": "mw-0c50858d01EXAMPLE",
               "WindowExecutionId": "14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE",
               "Status": "SUCCESS",
               "StartTime": 1557593793.483,
               "EndTime": 1557593798.978
           },
           {
               "WindowId": "mw-0c50858d01EXAMPLE",
               "WindowExecutionId": "791b72e0-f0da-4021-8b35-f95dfEXAMPLE",
               "Status": "SUCCESS",
               "StartTime": 1557593493.096,
               "EndTime": 1557593498.611
           },
           {
   ```

```
            "WindowId": "mw-0c50858d01EXAMPLE",
            "WindowExecutionId": "ecec60fa-6bb0-4d26-98c7-140308EXAMPLE",
            "Status": "SUCCESS",
            "StatusDetails": "No tasks to execute.",
            "StartTime": 1557593193.309,
            "EndTime": 1557593193.334
        }
    ]
}
```

2. Exécutez la commande suivante pour obtenir des informations sur l'exécution d'une tâche de fenêtre de maintenance.

   Linux

```
aws ssm get-maintenance-window-execution \
    --window-execution-id "14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE"
```

   Windows

```
aws ssm get-maintenance-window-execution ^
    --window-execution-id "14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE"
```

   Le système renvoie des informations similaires à celles qui suivent.

```
{
    "WindowExecutionId": "14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE",
    "TaskIds": [
        "c9b05aba-197f-4d8d-be34-e73fbEXAMPLE"
    ],
    "Status": "SUCCESS",
    "StartTime": 1557593493.096,
    "EndTime": 1557593498.611
}
```

3. Exécutez la commande suivante pour répertorier les tâches exécutées dans le cadre de l'exécution d'une fenêtre de maintenance.

   Linux

```
aws ssm describe-maintenance-window-execution-tasks \
    --window-execution-id "14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE"
```

   Windows

```
aws ssm describe-maintenance-window-execution-tasks ^
    --window-execution-id "14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE"
```

   Le système renvoie des informations similaires à celles qui suivent.

```
{
    "WindowExecutionTaskIdentities": [
        {
            "WindowExecutionId": "14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE",
            "TaskExecutionId": "c9b05aba-197f-4d8d-be34-e73fbEXAMPLE",
            "Status": "SUCCESS",
            "StartTime": 1557593493.162,
```

```
            "EndTime": 1557593498.57,
            "TaskArn": "AWS-RunShellScript",
            "TaskType": "RUN_COMMAND"
        }
    ]
}
```

4. Exécutez la commande suivante pour obtenir les détails d'une exécution de tâche.

   Linux

```
aws ssm get-maintenance-window-execution-task \
    --window-execution-id "14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE" \
    --task-id "c9b05aba-197f-4d8d-be34-e73fbEXAMPLE"
```

   Windows

```
aws ssm get-maintenance-window-execution-task ^
    --window-execution-id "14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE" ^
    --task-id "c9b05aba-197f-4d8d-be34-e73fbEXAMPLE"
```

   Le système renvoie des informations similaires à celles qui suivent.

```
{
    "WindowExecutionId": "14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE",
    "TaskExecutionId": "c9b05aba-197f-4d8d-be34-e73fbEXAMPLE",
    "TaskArn": "AWS-RunShellScript",
    "ServiceRole": "arn:aws:iam::111122223333:role/aws-service-role/ssm.amazonaws.com/
AWSServiceRoleForAmazonSSM",
    "Type": "RUN_COMMAND",
    "TaskParameters": [
        {
            "aws:InstanceId": {
                "Values": [
                    "i-02573cafcfEXAMPLE"
                ]
            },
            "commands": {
                "Values": [
                    "df"
                ]
            }
        }
    ],
    "Priority": 10,
    "MaxConcurrency": "1",
    "MaxErrors": "1",
    "Status": "SUCCESS",
    "StartTime": 1557593493.162,
    "EndTime": 1557593498.57
}
```

5. Exécutez la commande suivante pour obtenir les appels de tâche spécifiques exécutés pendant une exécution de tâche.

   Linux

```
aws ssm describe-maintenance-window-execution-task-invocations \
    --window-execution-id "14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE" \
    --task-id "c9b05aba-197f-4d8d-be34-e73fbEXAMPLE"
```

Windows

```
aws ssm describe-maintenance-window-execution-task-invocations ^
    --window-execution-id "14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE" ^
    --task-id "c9b05aba-197f-4d8d-be34-e73fbEXAMPLE"
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
    "WindowExecutionTaskInvocationIdentities": [
        {
            "WindowExecutionId": "14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE",
            "TaskExecutionId": "c9b05aba-197f-4d8d-be34-e73fbEXAMPLE",
            "InvocationId": "c336d2ab-09de-44ba-8f6a-6136cEXAMPLE",
            "ExecutionId": "76a5a04f-caf6-490c-b448-92c02EXAMPLE",
            "TaskType": "RUN_COMMAND",
            "Parameters": "{\"documentName\":\"AWS-RunShellScript\",\"instanceIds\":
[\"i-02573cafcfEXAMPLE\"],\"maxConcurrency\":\"1\",\"maxErrors\":\"1\",\"parameters\":
{\"commands\":[\"df\"]}}",
            "Status": "SUCCESS",
            "StatusDetails": "Success",
            "StartTime": 1557593493.222,
            "EndTime": 1557593498.466
        }
    ]
}
```

Didacticiel : Mettre à jour une fenêtre de maintenance (AWS CLI)

Ce didacticiel explique comment utiliser l'AWS CLI pour mettre à jour une fenêtre de maintenance. Il vous montre également comment mettre à jour différents types de tâche, notamment des tâches d'exécution de commande Systems Manager, d'automatisation Systems Manager, AWS Lambda et AWS Step Functions.

Les exemples de cette section utilisent les actions Systems Manager suivantes pour la mise à jour d'une fenêtre de maintenance.

- UpdateMaintenanceWindow
- UpdateMaintenanceWindowTarget
- UpdateMaintenanceWindowTask
- DeregisterTargetFromMaintenanceWindow

Pour des informations sur l'utilisation de la console Systems Manager afin de mettre à jour une fenêtre de maintenance, consultez Mettre à jour ou supprimer une fenêtre de maintenance (console) (p. 713).

Tout au long des étapes de ce didacticiel, remplacez les valeurs indiquées en italiques et en *rouge* par vos propres options et identifiants. Par exemple, remplacez l'ID de la fenêtre de maintenance *mw-0c50858d01EXAMPLE* et l'ID d' instance *i-02573cafcfEXAMPLE* par des ID de ressource que vous créez.

Pour mettre à jour une fenêtre de maintenance (AWS CLI)

1. Ouvrez l'AWS CLI et exécutez la commande suivante pour mettre à jour une cible afin d'inclure un nom et une description :

Linux

```
aws ssm update-maintenance-window-target \
    --window-id "mw-0c50858d01EXAMPLE" \
    --window-target-id "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE" \
    --name "My-Maintenance-Window-Target" \
    --description "Description for my maintenance window target"
```

Windows

```
aws ssm update-maintenance-window-target ^
    --window-id "mw-0c50858d01EXAMPLE" ^
    --window-target-id "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE" ^
    --name "My-Maintenance-Window-Target" ^
    --description "Description for my maintenance window target"
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
    "WindowId": "mw-0c50858d01EXAMPLE",
    "WindowTargetId": "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE",
    "Targets": [
        {
            "Key": "InstanceIds",
            "Values": [
                "i-02573cafcfEXAMPLE"
            ]
        }
    ],
    "Name": "My-Maintenance-Window-Target",
    "Description": "Description for my maintenance window target"
}
```

2. Exécutez la commande suivante pour utiliser l'option `replace` afin de supprimer le champ de description et d'ajouter une cible supplémentaire. Le champ de description est supprimé car la mise à jour ne l'inclut pas (valeur nulle). Assurez-vous de spécifier une instance supplémentaire qui a été configurée pour être utilisée avec Systems Manager :

   Linux

```
aws ssm update-maintenance-window-target \
    --window-id "mw-0c50858d01EXAMPLE" \
    --window-target-id "d208dedf-3f6b-41ff-ace8-8e751EXAMPLE" \
    --targets "Key=InstanceIds,Values=i-02573cafcfEXAMPLE,i-0471e04240EXAMPLE" \
    --name "My-Maintenance-Window-Target" \
    --replace
```

   Windows

```
aws ssm update-maintenance-window-target ^
    --window-id "mw-0c50858d01EXAMPLE" ^
    --window-target-id "d208dedf-3f6b-41ff-ace8-8e751EXAMPLE" ^
    --targets "Key=InstanceIds,Values=i-02573cafcfEXAMPLE,i-0471e04240EXAMPLE" ^
    --name "My-Maintenance-Window-Target" ^
    --replace
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
    "WindowId": "mw-0c50858d01EXAMPLE",
    "WindowTargetId": "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE",
    "Targets": [
        {
            "Key": "InstanceIds",
            "Values": [
                "i-02573cafcfEXAMPLE",
                "i-0471e04240EXAMPLE"
            ]
        }
    ],
    "Name": "My-Maintenance-Window-Target"
}
```

3. L'option `start-date` vous permet de retarder l'activation d'une fenêtre de maintenance jusqu'à une date ultérieure spécifiée. L'option `end-date` vous permet de définir une date et une heure futures après lesquelles la fenêtre de maintenance ne s'exécutera plus. Spécifiez les options au format ISO-8601 étendu.

   Exécutez la commande suivante pour spécifier une plage de dates et d'heures pour les exécutions de fenêtre de maintenance régulièrement planifiées.

   Linux

   ```
   aws ssm update-maintenance-window \
       --window-id "mw-0c50858d01EXAMPLE" \
       --start-date "2020-10-01T10:10:10Z" \
       --end-date "2020-11-01T10:10:10Z"
   ```

   Windows

   ```
   aws ssm update-maintenance-window ^
       --window-id "mw-0c50858d01EXAMPLE" ^
       --start-date "2020-10-01T10:10:10Z" ^
       --end-date "2020-11-01T10:10:10Z"
   ```

4. Exécutez la commande suivante pour mettre à jour une tâche Fonctionnalité Exécuter la commande.

   Tip

   Si votre cible est une instance EC2 pour Windows Server, remplacez `df` par `ipconfig` et `AWS-RunShellScript` par `AWS-RunPowerShellScript` dans la commande suivante.

   Linux

   ```
   aws ssm update-maintenance-window-task \
       --window-id "mw-0c50858d01EXAMPLE" \
       --window-task-id "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE" \
       --targets "Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE" \
       --task-arn "AWS-RunShellScript" \
       --service-role-arn "arn:aws:iam::111122223333:role/MaintenanceWindowsRole" \
       --task-invocation-parameters "RunCommand={Comment=Revising my Run Command
    task,Parameters={commands=df}}" \
       --priority 1 --max-concurrency 10 --max-errors 4 \
       --name "My-Task-Name" --description "A description for my Run Command task"
   ```

Windows

```
aws ssm update-maintenance-window-task ^
    --window-id "mw-0c50858d01EXAMPLE" ^
    --window-task-id "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE" ^
    --targets "Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE" ^
    --task-arn "AWS-RunShellScript" ^
    --service-role-arn "arn:aws:iam::111122223333:role/MaintenanceWindowsRole" ^
    --task-invocation-parameters "RunCommand={Comment=Revising my Run Command
 task,Parameters={commands=df}}" ^
    --priority 1 --max-concurrency 10 --max-errors 4 ^
    --name "My-Task-Name" --description "A description for my Run Command task"
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
    "WindowId": "mw-0c50858d01EXAMPLE",
    "WindowTaskId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE",
    "Targets": [
        {
            "Key": "WindowTargetIds",
            "Values": [
                "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
            ]
        }
    ],
    "TaskArn": "AWS-RunShellScript",
    "ServiceRoleArn": "arn:aws:iam::111122223333:role/MaintenanceWindowsRole",
    "TaskParameters": {},
    "TaskInvocationParameters": {
        "RunCommand": {
            "Comment": "Revising my Run Command task",
            "Parameters": {
                "commands": [
                    "df"
                ]
            }
        }
    },
    "Priority": 1,
    "MaxConcurrency": "10",
    "MaxErrors": "4",
    "Name": "My-Task-Name",
    "Description": "A description for my Run Command task"
}
```

5. Adaptez et exécutez la commande suivante pour mettre à jour une tâche Lambda.

   Linux

```
aws ssm update-maintenance-window-task \
    --window-id mw-0c50858d01EXAMPLE \
    --window-task-id 4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE \
    --targets "Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE" \
    --task-arn "arn:aws:lambda:us-east-2:111122223333:function:SSMTestLambda" \
    --service-role-arn "arn:aws:iam::111122223333:role/MaintenanceWindowsRole" \
    --task-invocation-parameters '{"Lambda":{"Payload":"{\"instanceId\":
\"{{RESOURCE_ID}}\",\"targetType\":\"{{TARGET_TYPE}}\"}"}}' \
    --priority 1 --max-concurrency 10 --max-errors 5 \
    --name "New-Lambda-Task-Name" \
```

```
    --description "A description for my Lambda task"
```

Windows

```
aws ssm update-maintenance-window-task ^
    --window-id mw-0c50858d01EXAMPLE ^
    --window-task-id 4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE ^
    --targets "Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE" ^
    --task-arn "arn:aws:lambda:us-east-2:111122223333:function:SSMTestLambda" ^
    --service-role-arn "arn:aws:iam::111122223333:role/MaintenanceWindowsRole" ^
    --task-invocation-parameters '{"Lambda":{"Payload":"{\"instanceId\":
\"{{RESOURCE_ID}}\",\"targetType\":\"{{TARGET_TYPE}}\"}"}}' ^
    --priority 1 --max-concurrency 10 --max-errors 5 ^
    --name "New-Lambda-Task-Name" ^
    --description "A description for my Lambda task"
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
    "WindowId": "mw-0c50858d01EXAMPLE",
    "WindowTaskId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE",
    "Targets": [
        {
            "Key": "WindowTargetIds",
            "Values": "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
        }
    ],
    "TaskArn": "arn:aws:lambda:us-east-2:111122223333:function:SSMTestLambda",
    "ServiceRoleArn": "arn:aws:iam::111122223333:role/MaintenanceWindowsRole",
    "TaskParameters": {},
    "TaskInvocationParameters": {
        "Lambda": {
            "Payload": "e30="
        }
    },
    "Priority": 1,
    "MaxConcurrency": "10",
    "MaxErrors": "5",
    "Name": "New-Lambda-Task-Name",
    "Description": "A description for my Lambda task"
}
```

6. Si vous mettez à jour une tâche AWS Step Functions, adaptez et exécutez la commande suivante pour mettre à jour ses paramètres d'invocation de tâche :

   Linux

```
aws ssm update-maintenance-window-task \
    --window-id "mw-0c50858d01EXAMPLE" \
    --window-task-id "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE" \
    --targets "Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE" \
    --task-arn "arn:aws:states:us-
east-2:111122223333:execution:SSMStepFunctionTest" \
    --service-role-arn "arn:aws:iam::111122223333:role/MaintenanceWindowsRole" \
    --task-invocation-parameters '{"StepFunctions":{"Input":"{\"instanceId\":
\"{{RESOURCE_ID}}\"}"}}' \
    --priority 0 --max-concurrency 10 --max-errors 5 \
    --name "My-Step-Functions-Task" \
    --description "A description for my Step Functions task"
```

Windows

```
aws ssm update-maintenance-window-task ^
    --window-id "mw-0c50858d01EXAMPLE" ^
    --window-task-id "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE" ^
    --targets "Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE" ^
    --task-arn "arn:aws:states:us-
east-2:111122223333:execution:SSMStepFunctionTest" ^
    --service-role-arn "arn:aws:iam::111122223333:role/MaintenanceWindowsRole" ^
    --task-invocation-parameters '{"StepFunctions":{"Input":"{\"instanceId\":
\"{{RESOURCE_ID}}\"}"}}' ^
    --priority 0 --max-concurrency 10 --max-errors 5 ^
    --name "My-Step-Functions-Task" ^
    --description "A description for my Step Functions task"
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
    "WindowId": "mw-0c50858d01EXAMPLE",
    "WindowTaskId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE",
    "Targets": [
        {
            "Key": "WindowTargetIds",
            "Values": [
                "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
            ]
        }
    ],
    "TaskArn": "arn:aws:states:us-east-2:111122223333:execution:SSMStepFunctionTest",
    "ServiceRoleArn": "arn:aws:iam::111122223333:role/MaintenanceWindowsRole",
    "TaskParameters": {},
    "TaskInvocationParameters": {
        "StepFunctions": {
            "Input": "{\"instanceId\":\"{{RESOURCE_ID}}\"}"
        }
    },
    "Priority": 0,
    "MaxConcurrency": "10",
    "MaxErrors": "5",
    "Name": "My-Step-Functions-Task",
    "Description": "A description for my Step Functions task"
}
```

7. Exécutez la commande suivante pour annuler l'enregistrement d'une cible depuis une fenêtre de maintenance. Cet exemple utilise le paramètre `safe` pour déterminer si la cible est référencée par une tâche et si son enregistrement peut être annulé en toute sécurité.

   Linux

```
aws ssm deregister-target-from-maintenance-window \
    --window-id "mw-0c50858d01EXAMPLE" \
    --window-target-id "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE" \
    --safe
```

   Windows

```
aws ssm deregister-target-from-maintenance-window ^
    --window-id "mw-0c50858d01EXAMPLE" ^
    --window-target-id "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE" ^
```

```
    --safe
```

Le système renvoie des informations similaires à celles qui suivent :

```
An error occurred (TargetInUseException) when calling the
 DeregisterTargetFromMaintenanceWindow operation:
This Target cannot be deregistered because it is still referenced in Task:
 4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE
```

8. Exécutez la commande suivante pour annuler l'enregistrement d'une cible depuis une fenêtre de maintenance même si la cible est référencée par une tâche. Vous pouvez forcer l'opération d'annulation d'enregistrement à l'aide du paramètre `no-safe`.

   Linux

   ```
   aws ssm deregister-target-from-maintenance-window \
       --window-id "mw-0c50858d01EXAMPLE" \
       --window-target-id "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE" \
       --no-safe
   ```

   Windows

   ```
   aws ssm deregister-target-from-maintenance-window ^
       --window-id "mw-0c50858d01EXAMPLE" ^
       --window-target-id "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE" ^
       --no-safe
   ```

   Le système renvoie des informations similaires à celles qui suivent :

   ```
   {
       "WindowId": "mw-0c50858d01EXAMPLE",
       "WindowTargetId": "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
   }
   ```

9. Exécutez la commande suivante pour mettre à jour une tâche Fonctionnalité Exécuter la commande. Cet exemple utilise un Systems Manager Parameter Store paramètre appelé `UpdateLevel` , qui est formaté comme suit : « ' »`{{ssm:UpdateLevel}}`

   Linux

   ```
   aws ssm update-maintenance-window-task \
       --window-id "mw-0c50858d01EXAMPLE" \
       --window-task-id "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE" \
       --targets "Key=InstanceIds,Values=i-02573cafcfEXAMPLE"  \
       --task-invocation-parameters "RunCommand={Comment=A comment for my task
    update,Parameters={UpdateLevel='{{ssm:UpdateLevel}}'}}"
   ```

   Windows

   ```
   aws ssm update-maintenance-window-task ^
       --window-id "mw-0c50858d01EXAMPLE" ^
       --window-task-id "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE" ^
       --targets "Key=InstanceIds,Values=i-02573cafcfEXAMPLE"  ^
       --task-invocation-parameters "RunCommand={Comment=A comment for my task
    update,Parameters={UpdateLevel='{{ssm:UpdateLevel}}'}}"
   ```

Le système renvoie des informations similaires à celles qui suivent :

```
{
    "WindowId": "mw-0c50858d01EXAMPLE",
    "WindowTaskId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE",
    "Targets": [
        {
            "Key": "InstanceIds",
            "Values": [
                "i-02573cafcfEXAMPLE"
            ]
        }
    ],
    "TaskArn": "AWS-RunShellScript",
    "ServiceRoleArn": "arn:aws:iam::111122223333:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
    "TaskParameters": {},
    "TaskInvocationParameters": {
        "RunCommand": {
            "Comment": "A comment for my task update",
            "Parameters": {
                "UpdateLevel": [
                    "{{ssm:UpdateLevel}}"
                ]
            }
        }
    },
    "Priority": 10,
    "MaxConcurrency": "1",
    "MaxErrors": "1"
}
```

10. Exécutez la commande suivante pour mettre à jour une tâche Automation afin qu'elle spécifie les paramètres WINDOW_ID et WINDOW_TASK_ID pour le paramètre `task-invocation-parameters` :

    Linux

    ```
    aws ssm update-maintenance-window-task \
        --window-id "mw-0c50858d01EXAMPLE" \
        --window-task-id "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE" \
        --targets "Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE \
        --task-arn "AutoTestDoc" \
        --service-role-arn arn:aws:iam::111122223333:role/aws-service-role/
    ssm.amazonaws.com/AWSServiceRoleForAmazonSSM \
        --task-invocation-parameters
     "Automation={Parameters={instanceId='{{RESOURCE_ID}}',initiator='{{WINDOW_ID}}.Task-
    {{WINDOW_TASK_ID}}'}}" \
        --priority 3 --max-concurrency 10 --max-errors 5
    ```

    Windows

    ```
    aws ssm update-maintenance-window-task ^
        --window-id "mw-0c50858d01EXAMPLE" ^
        --window-task-id "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE" ^
        --targets "Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE ^
        --task-arn "AutoTestDoc" ^
        --service-role-arn arn:aws:iam::111122223333:role/aws-service-role/
    ssm.amazonaws.com/AWSServiceRoleForAmazonSSM ^
```

```
    --task-invocation-parameters
 "Automation={Parameters={instanceId='{{RESOURCE_ID}}',initiator='{{WINDOW_ID}}.Task-
{{WINDOW_TASK_ID}}'}}" ^
    --priority 3 --max-concurrency 10 --max-errors 5
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
    "WindowId": "mw-0c50858d01EXAMPLE",
    "WindowTaskId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE",
    "Targets": [
        {
            "Key": "WindowTargetIds",
            "Values": [
                "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
            ]
        }
    ],
    "TaskArn": "AutoTestDoc",
    "ServiceRoleArn": "arn:aws:iam::111122223333:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
    "TaskParameters": {},
    "TaskInvocationParameters": {
        "Automation": {
            "Parameters": {
                "multi": [
                    "{{WINDOW_TASK_ID}}"
                ],
                "single": [
                    "{{WINDOW_ID}}"
                ]
            }
        }
    },
    "Priority": 0,
    "MaxConcurrency": "10",
    "MaxErrors": "5",
    "Name": "My-Automation-Task",
    "Description": "A description for my Automation task"
}
```

Didacticiel : Supprimer une fenêtre de maintenance (AWS CLI)

Pour supprimer une fenêtre de maintenance que vous avez créée dans ces didacticiels, exécutez la commande suivante :

```
aws ssm delete-maintenance-window --window-id "mw-0c50858d01EXAMPLE"
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
   "WindowId":"mw-0c50858d01EXAMPLE"
}
```

Procédures pas à pas d'une fenêtre de maintenance

Les procédures de cette section vous montrent comment créer une fenêtre de maintenance AWS Systems Manager à l'aide de l'AWS Command Line Interface ou de la console AWS Systems Manager. La fenêtre de maintenance que vous créez met à jour l'Agent SSM sur les instances gérées.

Table des matières

- Procédure : Créer une fenêtre de maintenance pour mettre à jour Agent SSM (AWS CLI) (p. 763)
- Procédure : Créer une fenêtre de maintenance à mettre à jour Agent SSM (console) (p. 768)

Vous pouvez également consulter des exemples de commandes dans la Référence relative à l'interface de ligne de commande AWS Systems Manager.

Procédure : Créer une fenêtre de maintenance pour mettre à jour Agent SSM (AWS CLI)

La procédure suivante vous montre comment utiliser l'AWS CLI pour créer une fenêtre de maintenance AWS Systems Manager. La procédure décrit également comment enregistrer vos instances gérées en tant que cibles et enregistrer une tâche Fonctionnalité Exécuter la commande pour mettre à jour l'Agent SSM.

Avant de commencer

Avant d'exécuter la procédure suivante, vous devez disposer de privilèges administrateur sur les instances que vous souhaitez configurer ou vous devez bénéficier des autorisations appropriées dans AWS Identity and Access Management (IAM). De plus, veillez à avoir au moins une instance EC2 pour Linux ou Windows Server en cours d'exécution et configurée pour Systems Manager. Pour plus d'informations, consultez la section Prérequis d'Systems Manager (p. 15).

Rubriques

- Étape 1 : Mise en route (p. 763)
- Étape 2 : Crée la fenêtre de maintenance (p. 764)
- Étape 3 : Enregistrer les cibles de fenêtre de maintenance (AWS CLI) (p. 765)
- Étape 4 : Enregistrer une Fonctionnalité Exécuter la commande tâche pour la fenêtre de maintenance à mettre à jourAgent SSM (p. 767)

Étape 1 : Mise en route

Pour exécuter des commandes à l'aide de l'AWS CLI

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).
2. Vérifiez qu'une instance est prête à être enregistrée en tant que cible d'une fenêtre de maintenance.

   Exécutez la commande suivante pour afficher les instances qui sont en ligne.

   ```
   aws ssm describe-instance-information --query "InstanceInformationList[*]"
   ```

   Exécutez la commande suivante pour afficher les détails sur une instance spécifique.

   ```
   aws ssm describe-instance-information --instance-information-filter-list
    key=InstanceIds,valueSet=instance-id
   ```

Étape 2 : Crée la fenêtre de maintenance

Utilisez la procédure suivante pour créer une fenêtre de maintenance et spécifier ses options de base, telles que la planification et la durée.

Créer une fenêtre de maintenance (AWS CLI)

1. Ouvrez l'AWS CLI et exécutez les commandes suivantes pour créer une fenêtre de maintenance qui s'exécute toutes les semaines le dimanche à 02 h 00, dans le fuseau horaire États-Unis - Pacifique, avec une coupure d'une heure :

   Linux

   ```
   aws ssm create-maintenance-window \
       --name "My-First-Maintenance-Window" \
       --schedule "cron(0 2 ? * SUN *)" \
       --duration 2 \
       --schedule-timezone "America/Los_Angeles" \
       --cutoff 1 \
       --no-allow-unassociated-targets
   ```

   Windows

   ```
   aws ssm create-maintenance-window ^
       --name "My-First-Maintenance-Window" ^
       --schedule "cron(0 2 ? * SUN *)" ^
       --duration 2 ^
       --schedule-timezone "America/Los_Angeles" ^
       --cutoff 1 ^
       --no-allow-unassociated-targets
   ```

   Pour de plus amples informations sur la création d'expressions cron pour le paramètre `schedule`, veuillez consulter Référence : Expressions cron et rate pour Systems Manager (p. 1392).

   Pour obtenir une explication des relations entre les différentes options liées à la planification pour les fenêtres de maintenance, consultez Référence : Options de planification de la fenêtre de maintenance et de période active (p. 772).

   Pour de plus amples informations sur l'utilisation de l'option `--schedule`, veuillez consulter Référence : Expressions cron et rate pour Systems Manager (p. 1392).

   Le système renvoie des informations similaires à celles qui suivent :

   ```
   {
      "WindowId":"mw-0c50858d01EXAMPLE"
   }
   ```

2. Pour répertorier les fenêtres de maintenance créées dans votre compte AWS dans votre région AWS actuelle, exécutez la commande suivante :

   ```
   aws ssm describe-maintenance-windows
   ```

   Le système renvoie des informations similaires à celles qui suivent :

   ```
   {
       "WindowIdentities": [
           {
   ```

```
            "Cutoff": 1,
            "Name": "My-First-Maintenance-Window",
            "NextExecutionTime": "2019-02-03T02:00-08:00",
            "Enabled": true,
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Duration": 2
        }
    ]
}
```

Étape 3 : Enregistrer les cibles de fenêtre de maintenance (AWS CLI)

Utilisez la procédure suivante pour enregistrer une cible avec votre fenêtre de maintenance créée à l'étape 2. En enregistrant une cible, vous spécifiez les instances à mettre à jour.

Pour enregistrer des cibles de fenêtre de maintenance (AWS CLI)

1. Exécutez la commande suivante :

   Linux

   ```
   aws ssm register-target-with-maintenance-window \
       --window-id "mw-0c50858d01EXAMPLE" \
       --target "Key=InstanceIds,Values=i-02573cafcfEXAMPLE" \
       --resource-type "INSTANCE"
   ```

   Windows

   ```
   aws ssm register-target-with-maintenance-window ^
       --window-id "mw-0c50858d01EXAMPLE" ^
       --target "Key=InstanceIds,Values=i-02573cafcfEXAMPLE" ^
       --resource-type "INSTANCE"
   ```

   Le système renvoie des informations similaires à celles qui suivent, qui incluent un ID de cible de fenêtre de maintenance. Copiez ou notez la WindowTargetId valeur. Vous devrez spécifier cet ID à l'étape suivante pour enregistrer une tâche pour cette fenêtre de maintenance.

   ```
   {
      "WindowTargetId":"1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
   }
   ```

   Autres commandes

   Utilisez la commande suivante pour enregistrer plusieurs instances gérées :

   Linux

   ```
   aws ssm register-target-with-maintenance-window \
       --window-id "mw-0c50858d01EXAMPLE" \
       --targets "Key=InstanceIds,Values=i-02573cafcfEXAMPLE,i-0471e04240EXAMPLE" \
       --resource-type "INSTANCE"
   ```

   Windows

   ```
   aws ssm register-target-with-maintenance-window ^
       --window-id "mw-0c50858d01EXAMPLE" ^
   ```

```
    --targets "Key=InstanceIds,Values=i-02573cafcfEXAMPLE,i-0471e04240EXAMPLE" ^
    --resource-type "INSTANCE"
```

Utilisez la commande suivante pour enregistrer les instances en utilisant des balises Amazon EC2. Exemple :

Linux

```
aws ssm register-target-with-maintenance-window \
    --window-id "mw-0c50858d01EXAMPLE" \
    --targets "Key=tag:Environment,Values=Prod" "Key=tag:Role,Values=Web" \
    --resource-type "INSTANCE"
```

Windows

```
aws ssm register-target-with-maintenance-window ^
    --window-id "mw-0c50858d01EXAMPLE" ^
    --targets "Key=tag:Environment,Values=Prod" "Key=tag:Role,Values=Web" ^
    --resource-type "INSTANCE"
```

2. Utilisez la commande suivante pour afficher les cibles d'une fenêtre de maintenance :

```
aws ssm describe-maintenance-window-targets --window-id "mw-0c50858d01EXAMPLE"
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
    "Targets": [
        {
            "ResourceType": "INSTANCE",
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Targets": [
                {
                    "Values": [
                        "i-02573cafcfEXAMPLE"
                    ],
                    "Key": "InstanceIds"
                }
            ],
            "WindowTargetId": "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
        },
        {
            "ResourceType": "INSTANCE",
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Targets": [
                {
                    "Values": [
                        "Prod"
                    ],
                    "Key": "tag:Environment"
                },
                {
                    "Values": [
                        "Web"
                    ],
                    "Key": "tag:Role"
                }
            ],
            "WindowTargetId": "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
        }
```

```
    ]
}
```

Étape 4 : Enregistrer une Fonctionnalité Exécuter la commande tâche pour la fenêtre de maintenance à mettre à jourAgent SSM

Utilisez la procédure suivante pour enregistrer une tâche Fonctionnalité Exécuter la commande pour la fenêtre de maintenance que vous avez créée à l'étape 2. La tâche Fonctionnalité Exécuter la commande met à jour l'Agent SSM sur les cibles enregistrées.

Pour enregistrer une tâche Fonctionnalité Exécuter la commande pour une fenêtre de maintenance à mettre à jour Agent SSM (AWS CLI)

1. Exécutez la commande suivante pour enregistrer une Fonctionnalité Exécuter la commande tâche pour la fenêtre de maintenance à l'aide de la WindowTargetId valeur de l'étape 3. La tâche met à jour l'Agent SSM en utilisant le document `AWS-UpdateSSMAgent`.

   Linux

   ```
   aws ssm register-task-with-maintenance-window \
       --window-id "mw-0c50858d01EXAMPLE" \
       --task-arn "AWS-UpdateSSMAgent" \
       --name "UpdateSSMAgent" \
       --targets "Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE" \
       --service-role-arn "arn:aws:iam::1122334455:role/MW-Role" \
       --task-type "RUN_COMMAND" \
       --max-concurrency 1 --max-errors 1 --priority 10
   ```

   Windows

   ```
   aws ssm register-task-with-maintenance-window ^
       --window-id "mw-0c50858d01EXAMPLE" ^
       --task-arn "AWS-UpdateSSMAgent" ^
       --name "UpdateSSMAgent" ^
       --targets "Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE" ^
       --service-role-arn "arn:aws:iam::1122334455:role/MW-Role" ^
       --task-type "RUN_COMMAND" ^
       --max-concurrency 1 --max-errors 1 --priority 10
   ```

   Note

   Si les cibles que vous avez enregistrées à l'étape précédente sont des cibles Windows Server 2012 R2 ou version antérieure, vous devez utiliser le document `AWS-UpdateEC2Config`.

   Le système renvoie des informations similaires à celles qui suivent :

   ```
   {
      "WindowTaskId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE"
   }
   ```

2. Exécutez la commande suivante pour afficher toutes les tâches enregistrées pour une fenêtre de maintenance.

   ```
   aws ssm describe-maintenance-window-tasks --window-id "mw-0c50858d01EXAMPLE"
   ```

Le système renvoie des informations similaires à celles qui suivent :

```
{
    "Tasks": [
        {
            "ServiceRoleArn": "arn:aws:iam::111122223333:role/MW-Role",
            "MaxErrors": "1",
            "TaskArn": "AWS-UpdateSSMAgent",
            "MaxConcurrency": "1",
            "WindowTaskId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE",
            "TaskParameters": {},
            "Priority": 10,
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Type": "RUN_COMMAND",
            "Targets": [
                {
                    "Values": [
                        "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
                    ],
                    "Key": "WindowTargetIds"
                }
            ],
            "Name": "UpdateSSMAgent"
        }
    ]
}
```

Procédure : Créer une fenêtre de maintenance à mettre à jour Agent SSM (console)

La procédure suivante vous montre comment utiliser la console AWS Systems Manager pour créer une fenêtre de maintenance AWS Systems Manager. La procédure décrit également comment enregistrer vos instances gérées en tant que cibles et enregistrer une tâche Fonctionnalité Exécuter la commande pour mettre à jour l'Agent SSM.

Avant de commencer

Avant d'exécuter la procédure suivante, vous devez disposer de privilèges administrateur sur les instances que vous souhaitez configurer ou vous devez bénéficier des autorisations appropriées dans AWS Identity and Access Management (IAM). De plus, veillez à avoir au moins une instance EC2 pour Linux ou Windows Server en cours d'exécution et configurée pour Systems Manager. Pour plus d'informations, consultez la section Prérequis d'Systems Manager (p. 15).

Rubriques

- Étape 1 : Créer la fenêtre de maintenance (console) (p. 768)
- Étape 2 : Enregistrement des cibles de fenêtre de maintenance (console) (p. 769)
- Étape 3 : Enregistrer une Fonctionnalité Exécuter la commande tâche pour la fenêtre de maintenance à mettre à jour Agent SSM (console) (p. 770)

Étape 1 : Créer la fenêtre de maintenance (console)

Créer une fenêtre de maintenance (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fenêtres de maintenance.

3. Choisissez Create maintenance window (Créer une fenêtre de maintenance).
4. Pour Name (Nom), tapez un nom descriptif pour vous aider à identifier cette fenêtre de maintenance comme fenêtre de maintenance de test.
5. Pour Description, entrez une description.
6. Choisissez Allow unregistered targets (Autoriser les cibles non enregistrées) si vous souhaitez autoriser une tâche de fenêtre de maintenance à exécuter des instances gérées, même si vous n'avez pas enregistré ces instances en tant que cibles. Si vous choisissez cette option, vous pouvez choisir des instances non enregistrées (par ID d'instance) lorsque vous enregistrez une tâche avec la fenêtre de maintenance.

   Si vous ne choisissez pas cette option, vous devez choisir des cibles enregistrées au préalable lorsque vous enregistrez une tâche avec la fenêtre de maintenance.
7. Spécifiez un programme pour la fenêtre de maintenance à l'aide d'une des trois options de programmation.

   Pour plus d'informations sur la génération d'expressions cron/rate, consultez Référence : Expressions cron et rate pour Systems Manager (p. 1392).
8. Pour Durée, entrez le nombre d'heures pendant lequel la fenêtre de maintenance doit s'exécuter.
9. Dans le champ Stop initiating tasks (Arrêter le lancement des tâches), entrez le nombre d'heures avant la fin de la fenêtre de maintenance pendant lequel le système doit cesser de planifier l'exécution de nouvelles tâches.
10. (Facultatif) Pour Start date (optional) [Date de début (facultatif)], spécifiez la date et l'heure, au format ISO-8601 étendu, où vous voulez que la fenêtre de maintenance devienne active. Cela vous permet de retarder l'activation de la fenêtre de maintenance jusqu'à la date ultérieure spécifiée.
11. (Facultatif) Pour Start date (optional) [Date de début (facultatif)], spécifiez la date et l'heure, au format ISO-8601 étendu, où vous voulez que la fenêtre de maintenance devienne inactive. Cela vous permet de définir une date et une heure futures après lesquelles la fenêtre de maintenance ne s'exécutera plus.
12. (Facultatif) Pour Time zone (optional) [Fuseau horaire (facultatif)], spécifiez le fuseau horaire sur lequel baser les exécutions de fenêtre de maintenance planifiées, au format IANA (Internet Assigned Numbers Authority). Exemple : « Amérique/Los_Angeles », « etc/UTC » ou « Asie/Séoul ».

    Pour plus d'informations sur les formats valides, consultez Time Zone Database sur le site web de l'IANA.
13. Choisissez Create maintenance window (Créer une fenêtre de maintenance). Le système vous renvoie à la page de la fenêtre de maintenance. La fenêtre de maintenance que vous venez de créer possède l'état Activé.

Étape 2 : Enregistrement des cibles de fenêtre de maintenance (console)

Utilisez la procédure suivante pour enregistrer une cible avec la fenêtre de maintenance que vous avez créée à l'étape 1. En enregistrant une cible, vous spécifiez les instances à mettre à jour.

Affecter des cibles à une fenêtre de maintenance (console)

1. Dans la liste des fenêtres de maintenance, sélectionnez la fenêtre de maintenance que vous venez de créer.
2. Choisissez Actions, puis Register targets (Enregistrer les cibles).
3. Pour Target Name (Nom de cible), entrez un nom pour la cible.
4. Pour Description, entrez une description.
5. (Facultatif) Pour Owner information (Informations sur le propriétaire), précisez votre nom ou alias de travail. Les informations sur le propriétaire sont incluses dans tout Amazon EventBridge événement déclenché lors de l'exécution de tâches pour ces cibles dans cette fenêtre de maintenance.

Pour de plus amples informations sur l'utilisation de EventBridge pour surveiller les événements Systems Manager, veuillez consulter Surveillance des événements Systems Manager avec Amazon EventBridge (p. 1340).

6. Dans la section Select targets by (Sélectionner les cibles par), choisissez Specifying Tags (Spécification de balises) pour cibler les instances en utilisant les balises Amazon EC2 que vous avez attribuées au préalable à ces instances. Sélectionnez Manually Selecting Instances pour choisir des instances individuelles en fonction de leur instance IDs .

   Note

   Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.

7. Choisissez Register target (Enregistrer la cible).

Étape 3 : Enregistrer une Fonctionnalité Exécuter la commande tâche pour la fenêtre de maintenance à mettre à jour Agent SSM (console)

Utilisez la procédure suivante pour enregistrer une tâche Fonctionnalité Exécuter la commande pour la fenêtre de maintenance que vous avez créée à l'étape 1. La tâche Fonctionnalité Exécuter la commande met à jour l'Agent SSM sur les cibles enregistrées.

Pour attribuer des tâches à une fenêtre de maintenance (console)

1. Dans la liste des fenêtres de maintenance, sélectionnez la fenêtre de maintenance que vous venez de créer.
2. Choisissez Actions, puis Register Fonctionnalité Exécuter la commande task (Enregistrer une tâche d'exécution de commande).
3. Pour Name (Nom), entrez un nom pour la tâche, par exemple .UpdateSSMAgent
4. Pour Description, entrez une description.
5. Pour Document, choisissez le document de commande SSM `AWS-UpdateSSMAgent`.

   Note

   Si les cibles que vous avez enregistrées à l'étape précédente sont des cibles Windows Server 2012 R2 ou version antérieure, vous devez utiliser le document `AWS-UpdateEC2Config`.

6. Pour Priorité de tâche, spécifiez la priorité de cette tâche. Zéro (`0`) est la priorité la plus élevée. Les tâches d'une fenêtre de maintenance sont planifiées par ordre de priorité des tâches qui ont la même priorité planifiée en parallèle.
7. Dans la section Cibles, identifiez les instances sur lesquelles vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant manuellement les instances ou en indiquant un groupe de ressources.

   Note

   Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.

8. (Facultatif) Pour Contrôle du débit :

   - Pour Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage d'instances sur lesquelles exécuter simultanément la commande.

> **Note**
>
> Si vous avez sélectionné des cibles en indiquant des balises appliquées à des instances gérées ou en indiquant des groupes de ressources AWS, et que vous n'êtes pas sûr du nombre d'instances ciblées, limitez le nombre d'instances pouvant exécuter simultanément le document en spécifiant un pourcentage.

- Pour Seuil d'erreur, spécifiez quand arrêter l'exécution de la commande sur d'autres instances après son échec sur un nombre ou un pourcentage d'instances. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les instances sur lesquelles la commande est en cours de traitement peuvent également envoyer des erreurs.

9. Pour IAM service role (Rôle de service IAM), choisissez l'une des options suivantes pour autoriser Systems Manager à exécuter des tâches sur vos instances cibles :

   - Créer et utiliser un rôle lié à un service pour Systems Manager

     Les rôles liés à un service offrent une manière sécurisée d'accorder des autorisations aux services AWS, car seul le service lié peut assumer un rôle lié à un service. De plus, AWS définit automatiquement les autorisations des rôles liés à un service, en fonction des actions que le service lié effectue en votre nom.

     > **Note**
     >
     > Si un rôle lié à un service a déjà été créé pour votre compte, choisissez Use the service-linked role for Systems Manager (Utiliser le rôle lié à un service pour Systems Manager).

   - Use a custom service role (Utiliser un rôle de service personnalisé)

     Vous pouvez créer un rôle de service personnalisé pour les tâches de fenêtre de maintenance si vous souhaitez utiliser des autorisations plus rigoureuses que celles fournies par le rôle lié à un service.

     Si vous avez besoin de créer un rôle de service personnalisé, veuillez consulter l'une des rubriques suivantes :

     - Contrôle de l'accès aux fenêtres de maintenance (console) (p. 692)
     - Contrôler l'accès aux fenêtres de maintenance (AWS CLI) (p. 697)
     - Contrôler l'accès aux fenêtres de maintenance (Outils pour Windows PowerShell) (p. 703)

   Pour choisir plus facilement d'utiliser un rôle de service personnalisé ou le rôle lié à un service Systems Manager avec une tâche de fenêtre de maintenance, consultez Dois-je utiliser un rôle lié à un service ou un rôle de service personnalisé pour exécuter des tâches de fenêtre de maintenance ? (p. 691).

10. (Facultatif) Dans Output options (Options de sortie), pour enregistrer la sortie de la commande dans un fichier, sélectionnez Enable writing to an S3 bucket (Autoriser l'écriture dans un compartiment S3) Entrez les noms de compartiment et de préfixe (dossier) dans les zones.

    > **Note**
    >
    > Les autorisations S3 qui accordent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour de plus amples informations, veuillez consulter Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). En outre, si le compartiment S3 spécifié se trouve dans un compte AWS différent, assurez-vous que le profil d'instance associé à l'instance dispose des autorisations nécessaires pour écrire dans ce compartiment.

    Pour diffuser la sortie vers un groupe de CloudWatch Logs journaux , cochez la case CloudWatch de sortie. Tapez le nom du groupe de journaux dans la zone.

11. Dans la section SNS Notifications (Notifications SNS), vous pouvez éventuellement activer l'envoi par Systems Manager de notifications sur les statuts de commande à l'aide de Amazon SNS. Si vous choisissez d'activer cette option, vous devez spécifier les informations suivantes :
    a. Le rôle IAM devant déclencher les notifications Amazon SNS.
    b. La rubrique Amazon SNS à utiliser.
    c. Les types d'événement spécifiques dont vous souhaitez être averti.
    d. Le type de notification que vous souhaitez recevoir lors du changement de statut d'une commande. Pour les commandes envoyées à plusieurs instances, sélectionnez Invocation (Appel) pour recevoir une notification par appel (par instance) lors du changement de statut de chaque appel.
12. Dans la section Input Parameters (Paramètres d'entrée), vous pouvez éventuellement fournir une version spécifique de l'Agent SSM à installer, ou vous pouvez autoriser la rétrogradation du service Agent SSM à une version antérieure. Toutefois, pour cette procédure pas à pas, nous n'indiquons aucune version. Par conséquent, Agent SSM est mis à jour vers la dernière version.
13. Choisissez Register Fonctionnalité Exécuter la commande task (Enregistrer une tâche d'exécution de commande).

Référence : Options de planification de la fenêtre de maintenance et de période active

Lorsque vous créez une fenêtre de maintenance, vous devez spécifier la fréquence à laquelle celle-ci s'exécute à l'aide d'une expression Cron ou Rate (p. 1392). Si vous le souhaitez, vous pouvez spécifier une plage de dates pendant laquelle la fenêtre de maintenance peut s'exécuter conformément à une planification régulière, ainsi qu'un fuseau horaire sur lequel baser cette dernière.

Vous devez cependant savoir que l'option de fuseau horaire et les options de date de début/date de fin n'ont aucune influence les unes sur les autres. Toutes les dates de début et de fin que vous spécifiez (avec ou sans décalage par rapport à votre fuseau horaire) déterminent uniquement la période de validité pendant laquelle la fenêtre de maintenance peut s'exécuter aux dates et heures prévues. Une option de fuseau horaire détermine le fuseau horaire international sur lequel est basée la planification de la fenêtre de maintenance pendant sa période de validité.

> Note
>
> Vous spécifiez les dates de début et de fin au format d'horodatage ISO 8601. Par exemple: `2021-04-07T14:29:00-08:00`
> Vous spécifiez les fuseaux horaires au format IANA (Internet Assigned Numbers Authority). Exemple : `America/Chicago`, `Europe/Berlin` ou`Asia/Tokyo`

Exemples

- Exemple 1 : Spécifiez une date de début de fenêtre de maintenance (p. 772)
- Exemple 2 : Spécifiez une date de début et une date de fin de fenêtre de maintenance (p. 773)
- Exemple 3 : Crée une fenêtre de maintenance qui ne s'exécute qu'une seule fois. (p. 774)
- Exemple 4 : Spécifier le nombre de jours de décalage programmé pour une fenêtre de maintenance (p. 775)

Exemple 1 : Spécifiez une date de début de fenêtre de maintenance

Imaginons que vous utilisiez l’AWS CLI pour créer une fenêtre de maintenance avec les options suivantes :

- `--start-date 2021-01-01T00:00:00-08:00`
- `--schedule-timezone "America/Los_Angeles"`
- `--schedule "cron(0 09 ? * WED *)"`

Exemple :

Linux

```
aws ssm create-maintenance-window \
    --name "My-LAX-Maintenance-Window" \
    --allow-unassociated-targets \
    --duration 3 \
    --cutoff 1 \
    --start-date 2021-01-01T00:00:00-08:00 \
    --schedule-timezone "America/Los_Angeles" \
    --schedule "cron(0 09 ? * WED *)"
```

Windows

```
aws ssm create-maintenance-window ^
    --name "My-LAX-Maintenance-Window" ^
    --allow-unassociated-targets ^
    --duration 3 ^
    --cutoff 1 ^
    --start-date 2021-01-01T00:00:00-08:00 ^
    --schedule-timezone "America/Los_Angeles" ^
    --schedule "cron(0 09 ? * WED *)"
```

Cela signifie que la première exécution de la fenêtre de maintenance ne aura lieu qu'après la date et l'heure de début spécifiées, c'est-à-dire le vendredi 1er janvier 2021 à minuit, heure des États-Unis/Pacifique. (Ce fuseau horaire a huit heures de retard par rapport à l'heure UTC.) Notez que dans ce cas, la date et l'heure de début de la période de fenêtre ne représentent pas le moment où les fenêtres de maintenance s'exécutent pour la première fois. Si l'on considère les deux ensembles, les `--schedule-timezone` valeurs `--schedule` et signifient que la fenêtre de maintenance s'exécute à 9 heures tous les mercredis dans le fuseau horaire États-Unis/Pacifique (représenté par « Amérique/Los Angeles » au format IANA). La première exécution durant la période activée aura lieu le mercredi 4 janvier 2021, à 9 heures, heure des États-Unis/Pacifique.

Exemple 2 : Spécifiez une date de début et une date de fin de fenêtre de maintenance

Supposons que vous créiez une fenêtre de maintenance avec les options suivantes :

- `--start-date 2019-01-01T00:03:15+09:00`
- `--end-date 2019-06-30T00:06:15+09:00`
- `--schedule-timezone "Asia/Tokyo"`
- `--schedule "rate(7 days)"`

Exemple :

Linux

```
aws ssm create-maintenance-window \
    --name "My-NRT-Maintenance-Window" \
```

```
    --allow-unassociated-targets \
    --duration 3 \
    --cutoff 1 \
    --start-date 2019-01-01T00:03:15+09:00 \
    --end-date 2019-06-30T00:06:15+09:00 \
    --schedule-timezone "Asia/Tokyo" \
    --schedule "rate(7 days)"
```

Windows

```
aws ssm create-maintenance-window ^
    --name "My-NRT-Maintenance-Window" ^
    --allow-unassociated-targets ^
    --duration 3 ^
    --cutoff 1 ^
    --start-date 2019-01-01T00:03:15+09:00 ^
    --end-date 2019-06-30T00:06:15+09:00 ^
    --schedule-timezone "Asia/Tokyo" ^
    --schedule "rate(7 days)"
```

La période activée pour cette fenêtre de maintenance commence à 15 h 15, heure normale du Japon, le 1er janvier 2019. La période de validité de cette fenêtre de maintenance se termine à 6 h 15, heure normale du Japon, le dimanche 30 juin 2019. (Ce fuseau horaire a neuf heures d'avance par rapport à l'heure UTC.) Si l'on considère les deux ensembles, les `--schedule-timezone` valeurs `--schedule` et signifient que la fenêtre de maintenance s'exécute à 3 h 15 tous les mardis dans le fuseau horaire standard du Japon (représenté par « Asia/Tokyo » au format IANA). Cela est dû au fait que la fenêtre de maintenance s'exécute tous les sept jours et qu'elle devient active à 3 h 15 le mardi 1er janvier. La dernière exécution aura lieu à 3 h 15, heure normale du Japon, le mardi 25 juin 2019. Il s'agit du dernier mardi avant que la période de la fenêtre de maintenance activée se termine cinq jours plus tard.

Exemple 3 : Crée une fenêtre de maintenance qui ne s'exécute qu'une seule fois.

Vous avez à présent créé une fenêtre de maintenance avec cette option :

- `--schedule "at(2020-07-07T15:55:00)"`

Exemple :

Linux

```
aws ssm create-maintenance-window \
    --name "My-One-Time-Maintenance-Window" \
    --schedule "at(2020-07-07T15:55:00)" \
    --duration 5 \
    --cutoff 2 \
    --allow-unassociated-targets
```

Windows

```
aws ssm create-maintenance-window ^
    --name "My-One-Time-Maintenance-Window" ^
    --schedule "at(2020-07-07T15:55:00)" ^
    --duration 5 ^
    --cutoff 2 ^
    --allow-unassociated-targets
```

Cette fenêtre de maintenance s'exécute une seule fois, à 15 h 55 (UTC), le 7 juillet 2020. La fenêtre de maintenance est activée pour s'exécuter pendant près de cinq heures, selon les besoins, mais aucune nouvelle tâche ne peut démarrer deux heures avant la fin de la période de la fenêtre de maintenance.

Exemple 4 : Spécifier le nombre de jours de décalage programmé pour une fenêtre de maintenance

Vous avez à présent créé une fenêtre de maintenance avec cette option :

```
--schedule-offset 2
```

Exemple :

Linux

```
aws ssm create-maintenance-window \
    --name "My-Cron-Offset-Maintenance-Window" \
    --schedule "cron(0 30 23 ? * TUE#3 *)" \
    --duration 4 \
    --cutoff 1 \
    --schedule-offset 2 \
    --allow-unassociated-targets
```

Windows

```
aws ssm create-maintenance-window ^
    --name "My-Cron-Offset-Maintenance-Window" ^
    --schedule "cron(0 30 23 ? * TUE#3 *)" ^
    --duration 4 ^
    --cutoff 1 ^
    --schedule-offset 2 ^
    --allow-unassociated-targets
```

Un décalage de planification est le nombre de jours à attendre après la date et l'heure spécifiées par une expression CRON avant d'exécuter la fenêtre de maintenance.

Dans l'exemple ci-dessus, l'expression CRON planifie une fenêtre de maintenance à exécuter le troisième mardi de chaque mois à 23h30 :

```
--schedule "cron(0 30 23 ? * TUE#3 *)
```

Cependant, l'inclusion de `--schedule-offset 2` signifie que la fenêtre de maintenance ne sera pas exécutée avant 23h30 deux jours après le troisième mardi de chaque mois.

Les décalages de planification sont pris en charge uniquement pour les expressions CRON.

Contenu connexe

- Référence : Expressions cron et rate pour Systems Manager (p. 1392)
- Créer une fenêtre de maintenance (console) (p. 707)
- Didacticiel : Créer et configurer une fenêtre de maintenance (AWS CLI) (p. 717)
- CreateMaintenanceWindow dans le Référence d'API AWS Systems Manager.
- create-maintenance-window dans le AWS Systems Manager section of the AWS CLI Command Reference.
- Time Zone Database sur le site web de l'IANA

Fenêtres de maintenance de dépannage

Utilisez les informations suivantes pour résoudre les problèmes liés aux fenêtres de maintenance.

Rubriques

- Erreur de modification de tâche : Sur la page permettant de modifier une tâche de fenêtre de maintenance, la liste des IAM rôles renvoie un message d'erreur : « We couldn't find the IAM maintenance window role specified for this task. It might have been deleted, or it might not have been created yet. » (Nous n'avons pas pu trouver le rôle de fenêtre de maintenance IAM spécifié pour cette tâche. Il a peut-être été supprimé ou il n'a peut-être pas encore été créé.) (p. 776)
- Certaines cibles de fenêtre de maintenance ne sont pas mises à jour (p. 776)

Erreur de modification de tâche : Sur la page permettant de modifier une tâche de fenêtre de maintenance, la liste des IAM rôles renvoie un message d'erreur : « We couldn't find the IAM maintenance window role specified for this task. It might have been deleted, or it might not have been created yet. » (Nous n'avons pas pu trouver le rôle de fenêtre de maintenance IAM spécifié pour cette tâche. Il a peut-être été supprimé ou il n'a peut-être pas encore été créé.)

Problème 1 : Le rôle IAM de fenêtre de maintenance que vous avez spécifié initialement a été supprimé après avoir créé la tâche.

Correctifs possibles : (1) Sélectionnez un autre rôle IAM de fenêtre de maintenance, s'il existe pour votre compte, ou créez-en un nouveau et sélectionnez-le pour la tâche. (2) Créez ou sélectionnez un rôle lié à un service Systems Manager. Pour plus d'informations, consultez la section Dois-je utiliser un rôle lié à un service ou un rôle de service personnalisé pour exécuter des tâches de fenêtre de maintenance ? (p. 691).

Problème 2 : Si la tâche a été créée à l'aide de l' AWS CLI , des ou d'un kit SDK AWS, un nom de rôle Outils pour Windows PowerShell de fenêtre de maintenance inexistant pourrait avoir été spécifié.IAM Par exemple, il est possible que le rôle de fenêtre de maintenance de IAM ait été supprimé avant de créer la tâche, ou que le nom de rôle ait été saisi de façon incorrecte, par exemple **`myrole`** au lieu de **`my-role`**.

Correctifs possibles : (1) Sélectionnez le nom correct du rôle de fenêtre IAM de maintenance que vous souhaitez utiliser, ou créez-en un nouveau à spécifier pour la tâche. (2) Créez ou sélectionnez un rôle lié à un service Systems Manager. Pour plus d'informations, consultez la section Dois-je utiliser un rôle lié à un service ou un rôle de service personnalisé pour exécuter des tâches de fenêtre de maintenance ? (p. 691).

Certaines cibles de fenêtre de maintenance ne sont pas mises à jour

Problème : Vous remarquez que les tâches de fenêtre de maintenance n'ont pas été exécutées sur toutes les ressources ciblées par votre fenêtre de maintenance. Par exemple, dans les résultats d'exécution de la fenêtre de maintenance, la tâche pour cette ressource est marquée comme ayant échoué ou ayant expiré.

- Solution: Les raisons les plus courantes pour une tâche de fenêtre de maintenance qui ne s'exécute pas sur une ressource cible impliquent la connectivité et la disponibilité. Exemple :

- Systems Managera perdu la connexion à la ressource avant ou pendant l'opération de fenêtre de maintenance.
- La ressource a été hors ligne ou arrêtée pendant l'opération de fenêtre de maintenance.

Vous pouvez attendre la prochaine fenêtre de maintenance planifiée pour exécuter des tâches sur les ressources. Vous pouvez exécuter manuellement les tâches de la fenêtre de maintenance sur les ressources qui n'étaient pas disponibles ou qui étaient hors ligne.

Instances et nœuds deAWS Systems Manager

AWS Systems Manager fournit les fonctionnalités suivantes pour gérer vos instances EC2, vos serveurs sur site et les machines virtuelles de votre environnement hybride, ainsi que d'autres types de ressources AWS (nœuds).

Rubriques

- Conformité de la configuration AWS Systems Manager (p. 778)
- Inventaire AWS Systems Manager (p. 791)
- Instances gérées AWS Systems Manager (p. 851)
- Activations hybrides AWS Systems Manager (p. 868)
- AWS Systems Manager Session Manager (p. 869)
- AWS Systems Manager Fonctionnalité Exécuter la commande (p. 941)
- AWS Systems Manager State Manager (p. 987)
- AWS Systems Manager Gestionnaire de correctifs (p. 1047)
- AWS Systems Manager Distributor (p. 1174)

Conformité de la configuration AWS Systems Manager

Vous pouvez utiliser le service Conformité de configuration AWS Systems Manager pour analyser votre parc d'instances gérées afin de rechercher des incohérences de conformité et de configuration de correctifs. Vous pouvez collecter et regrouper des données de plusieurs comptes et régions AWS, puis explorer plus en détails des ressources spécifiques qui ne sont pas conformes. Par défaut, le service Conformité de configuration affiche les données de conformité actuelles relatives Systems Manager Gestionnaire de correctifs aux correctifs et Systems Manager State Manager aux associations.

Les données de conformité des correctifs du Gestionnaire de correctifs peuvent être envoyées à AWS Security Hub. Security Hub vous offre une vue complète des alertes de sécurité hautement prioritaires et de votre statut de conformité. Il surveille également le statut de l'application des correctifs de votre flotte. Pour de plus amples informations, veuillez consulter Intégration d'Gestionnaire de correctifs à AWS Security Hub (p. 1145).

La conformité Systems Manager offre les avantages et fonctions supplémentaires suivants :

- Affichez le suivi des modifications et l'historique de conformité pour les données et Gestionnaire de correctifs associations d'State Managerapplication de correctifs à l'aide d' AWS Config .
- Personnalisation de la conformité Systems Manager pour créer vos propres types de conformité en fonction de vos exigences métier ou informatiques.
- Correction des problèmes à l'aide de la fonctionnalité Systems Manager Fonctionnalité Exécuter la commande, de State Manager ou d'Amazon EventBridge.
- Transfert des données vers Amazon Athena et Amazon QuickSight pour générer des rapports à l'échelle du parc.

Le service Conformité de configuration est fourni sans frais supplémentaires. Vous payez uniquement pour les ressources AWS que vous utilisez.

Prise en charge de Amazon EventBridge

Cette fonction Systems Manager est prise en charge en tant que type d'événement dans les règles EventBridge. Pour plus d'informations, consultez Surveillance des événements Systems Manager avec Amazon EventBridge (p. 1340) et Référence : modèles et types Amazon EventBridge d'événements pourSystems Manager (p. 1387).

Intégration de ChefInSpec

Systems Managers'intègre à Chef InSpec . InSpec est une infrastructure d'exécution open source qui vous permet de créer des profils lisibles par l'utilisateur sur GitHub ou Amazon S3 . Ensuite, vous pouvez utiliser Systems Manager pour exécuter des analyses de conformité et afficher les instances conformes et non conformes. Pour plus d'informations, consultez la section Utilisation des InSpec profils Chef avec Systems Manager la conformité (p. 145).

Pricing

Le service Conformité de configuration est fourni sans frais supplémentaires. Vous payez uniquement pour les ressources AWS que vous utilisez.

Table des matières

- Mise en route avec le service Conformité de configuration (p. 779)
- Création d'une synchronisation de données de ressources pour Conformité de configuration (p. 780)
- Utilisation de la conformité de configuration (p. 782)
- Correction des problèmes de conformité à l'aide d'EventBridge (p. 785)
- Procédure pas à pas de Conformité de la configuration (AWS CLI) (p. 787)

Mise en route avec le service Conformité de configuration

Pour commencer à utiliser le service Conformité de configuration, effectuez les tâches suivantes.

Tâche	Pour plus d'informations, consultez
La conformité de configuration fonctionne avec les données de Gestionnaire de correctifs correctifs, State Manager les associations et les types de conformité personnalisée sur les instances Systems Manager gérées. Vérifiez que vos instances EC2 et vos machines hybrides (instances locales et machines virtuelles) sont configurées en tant qu'instances gérées en vérifiant les conditions préalables Systems Manager.	Prérequis d'Systems Manager (p. 15)
Mettez à jour Agent SSM sur vos instances gérées avec la dernière version.	Utilisation des Agent SSM (p. 72)
Si vous prévoyez de surveiller la conformité des correctifs, vérifiez que vous avez configuré Systems Manager Gestionnaire de correctifs . Vous devez effectuer des opérations d'application de correctifs à l'aide d' Gestionnaire de correctifs avant que le service Conformité de configuration puisse afficher les données de conformité des correctifs.	AWS Systems Manager Gestionnaire de correctifs (p. 1047)

Tâche	Pour plus d'informations, consultez
Si vous envisagez de surveiller la conformité des associations, vérifiez que vous avez créé des associations State Manager. Vous devez créer des associations pour que Conformité de configuration puisse afficher les données de conformité des associations.	AWS Systems Manager State Manager (p. 987)
(Facultatif) Configurez le système pour afficher le suivi des modifications et l'historique de conformité.	Affichage du suivi des modifications et de l'historique de configuration de la conformité (p. 785)
(Facultatif) Créez des types de conformité personnalisée.	Procédure pas à pas de Conformité de la configuration (AWS CLI) (p. 787)
(Facultatif) Créez une synchronisation de données de ressources pour regrouper toutes les données de conformité dans un compartiment S3 cible.	Création d'une synchronisation de données de ressources pour Conformité de configuration (p. 780)

Création d'une synchronisation de données de ressources pour Conformité de configuration

Vous pouvez utiliser la synchronisation de données de ressources Systems Manager pour envoyer les données de conformité depuis toutes vos instances gérées vers un compartiment S3 cible. Lors de la création de la synchronisation, vous pouvez spécifier les instances gérées à partir de plusieurs comptes et régions AWS, et votre environnement hybride sur site. La synchronisation de données de ressources met alors automatiquement à jour les données centralisées lors de la collecte de nouvelles données de conformité. Avec toutes les données de conformité stockées dans un compartiment S3 cible, vous pouvez utiliser les services tels qu'Amazon Athena et Amazon QuickSight pour interroger et analyser les données regroupées. La configuration de la synchronisation de données de ressources pour la conformité de configuration est une opération unique.

Le graphique suivant montre comment la synchronisation de données de ressources regroupe toutes les données de différents comptes, régions et votre environnement hybride dans un référentiel centralisé.

Utilisez la procédure suivante pour créer une synchronisation de données de ressources pour Conformité de configuration en utilisant la console Amazon EC2.

Pour créer et configurer un compartiment S3 pour la synchronisation de données de ressources (console)

1. Ouvrez la console Amazon S3 à l'adresse https://console.aws.amazon.com/s3/.
2. Créez un compartiment pour stocker vos données d'inventaire rassemblées. Pour de plus amples informations, veuillez consulter Créer un compartiment dans le Amazon Simple Storage Service Guide de mise en route. Notez le nom du compartiment et la région AWS dans laquelle vous l'avez créé.
3. Ouvrez le compartiment, sélectionnez l'onglet Autorisations, puis choisissez Stratégie de compartiment.
4. Copiez et collez la stratégie de compartiment suivante dans l'éditeur de stratégie. Remplacez *DOC-EXAMPLE-BUCKET* et *Account-ID* par le nom du compartiment S3 que vous avez créé et un ID de compte AWS valide. (Facultatif) Remplacez *Bucket-Prefix* par le nom d'un Amazon S3 préfixe (sous-répertoire). Si vous n'avez pas créé de préfixe, supprimez *Bucket-Prefix*/ à partir de l'ARN de la stratégie.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "SSMBucketPermissionsCheck",
            "Effect": "Allow",
            "Principal": {
                "Service": "ssm.amazonaws.com"
            },
            "Action": "s3:GetBucketAcl",
            "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET"
        },
        {
            "Sid": " SSMBucketDelivery",
            "Effect": "Allow",
            "Principal": {
                "Service": "ssm.amazonaws.com"
```

```
            },
            "Action": "s3:PutObject",
            "Resource": ["arn:aws:s3:::DOC-EXAMPLE-BUCKET/Bucket-Prefix/*/
accountid=Account_ID_number/*"],
            "Condition": {
                "StringEquals": {
                    "s3:x-amz-acl": "bucket-owner-full-control"
                }
            }
        }
    ]
}
```

Pour créer une synchronisation de données de ressources

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Instances gérées.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Instances gérées.
3. Choisissez Configure Inventory (Configurer l'inventaire), Resource Data Syncs (Synchronisations de données de ressources), puis Create resource data sync (Créer une synchronisation des données de ressource).
4. Dans le champ Nom de la synchronisation, entrez un nom pour la configuration de la synchronisation.
5. Dans le champ Nom du compartiment, entrez le nom du compartiment Amazon S3 créé au début de cette procédure.
6. (Facultatif) Dans le champ Préfixe du compartiment, tapez le nom d'un préfixe de compartiment S3 (sous-répertoire).
7. Dans le champ Région du compartiment, choisissez Cette région si le compartiment S3 créé est localisé dans la région AWS actuelle. Si le compartiment est localisé dans une autre région AWS, choisissez Autre région, et tapez le nom de la région.

   Note

   Si la synchronisation et le compartiment S3 cible sont localisés dans des régions différentes, vous pourriez être sujet à une tarification de transfert de données. Pour de plus amples informations, veuillez consulter Tarification Amazon S3.
8. Sélectionnez Create.

Utilisation de la conformité de configuration

Le service Conformité de configuration collecte et présente des données sur l'état d'Gestionnaire de correctifsapplication des correctifs, State Manager les associations et les types de conformité personnalisés. Cette section inclut des détails sur chacun de ces types de conformité et la manière d'afficher les données de conformité Systems Manager. Cette section inclut également des informations sur la façon d'afficher le suivi des modifications et l'historique de conformité.

Note

Systems Managers'intègre à Chef InSpec . InSpec est une infrastructure d'exécution open source qui vous permet de créer des profils lisibles par l'utilisateur sur GitHub ou Amazon S3 . Ensuite, vous pouvez utiliser Systems Manager pour exécuter des analyses de conformité et afficher les

instances conformes et non conformes. Pour plus d'informations, consultez la section Utilisation des InSpec profils Chef avec Systems Manager la conformité (p. 145).

Rubriques

- A propos de la conformité des correctifs (p. 783)
- A propos de la conformité des associations State Manager (p. 783)
- A propos de la conformité personnalisée (p. 783)
- Affichage des données de conformité actuelles (p. 784)
- Affichage du suivi des modifications et de l'historique de configuration de la conformité (p. 785)

A propos de la conformité des correctifs

Une fois que vous avez utilisé Systems Manager Gestionnaire de correctifs pour installer des correctifs sur vos instances, les informations relatives au statut de conformité sont immédiatement disponibles dans la console ou en réponse à AWS CLI des commandes de lSystems Manager' ou aux actions d'API correspondantes.

Pour de plus amples informations sur les valeurs du statut de conformité des correctifs, veuillez consulter Présentation des valeurs d'état de conformité des correctifs (p. 1113).

A propos de la conformité des associations State Manager

Une fois que vous avez créé une ou plusieurs associations State Manager, les informations relatives au statut de conformité sont immédiatement disponibles dans la console, ou en réponse à des commandes de l'AWS CLI ou aux actions d'API Systems Manager correspondantes. Pour les associations, le service Conformité de configuration affiche les statuts `Compliant` ou `Non-compliant` et le niveau de gravité attribué à l'association, tel que `Critical` ou `Medium`.

A propos de la conformité personnalisée

Vous pouvez attribuer des métadonnées de conformité à une instance gérée. Ces métadonnées peuvent ensuite être regroupées avec d'autres données de conformité à des fins de génération de rapports sur la conformité. Supposons par exemple que votre entreprise exécute les versions 2.0, 3.0 et 4.0 du logiciel X sur vos instances gérées. L'entreprise veut procéder à une normalisation dans la version 4.0, ce qui signifie que les instances qui exécutent les versions 2.0 et 3.0 ne sont pas conformes. Vous pouvez utiliser l'action PutComplianceItems d'API pour noter explicitement quelles instances gérées exécutent des versions antérieures du logiciel X. Actuellement, vous pouvez uniquement attribuer des métadonnées de conformité à l'aide de l' AWS CLI , des ou de l' Outils AWS pour Windows PowerShell .SDKs L'exemple de commande d'interface de ligne de commande suivant attribue des métadonnées de conformité à une instance gérée et spécifie le type de conformité au format requis`Custom:`.

Linux

```
aws ssm put-compliance-items \
    --resource-id i-1234567890abcdef0 \
    --resource-type ManagedInstance \
    --compliance-type Custom:SoftwareXCheck \
    --execution-summary ExecutionTime=AnyStringToDenoteTimeOrDate \
    --items Id=Version2.0,Title=SoftwareXVersion,Severity=CRITICAL,Status=NON_COMPLIANT
```

Windows

```
aws ssm put-compliance-items ^
    --resource-id i-1234567890abcdef0 ^
    --resource-type ManagedInstance ^
    --compliance-type Custom:SoftwareXCheck ^
```

```
    --execution-summary ExecutionTime=AnyStringToDenoteTimeOrDate ^
    --items Id=Version2.0,Title=SoftwareXVersion,Severity=CRITICAL,Status=NON_COMPLIANT
```

Les gestionnaires de la conformité peuvent ensuite afficher des récapitulatifs ou créer des rapports sur les instances conformes ou non. Vous pouvez attribuer au maximum 10 types différents de conformité personnalisée à une instance.

Pour obtenir un exemple montrant comment créer un type de conformité personnalisée et afficher les données de conformité, consultez Procédure pas à pas de Conformité de la configuration (AWS CLI) (p. 787).

Affichage des données de conformité actuelles

Cette section explique comment afficher les données de conformité dans la console AWS Systems Manager et à l'aide de l'AWS CLI. Pour de plus amples informations sur l'affichage du suivi des modifications, ainsi que de l'historique de conformité des associations et des correctifs, veuillez consulter Affichage du suivi des modifications et de l'historique de configuration de la conformité (p. 785).

Rubriques

- Affichage des données de conformité actuelles (console) (p. 784)
- Affichage des données de conformité actuelles (AWS CLI) (p. 784)

Affichage des données de conformité actuelles (console)

Utilisez la procédure suivante pour afficher les données de conformité dans la console Systems Manager.

Pour afficher les rapports de conformité actuelle dans la console Systems Manager

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Compliance (Conformité).

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Compliance (Conformité) dans le panneau de navigation.
3. Dans la zone Vue d'ensemble des détails pour les ressources, choisissez l'ID d'une instance.
4. Sur la page de détails de l'ID d'instance sélectionnez l'onglet Conformité de la configuration pour afficher son rapport détaillé de conformité de la configuration.

   Note

   Pour de plus amples informations sur la résolution des problèmes de conformité, veuillez consulter Correction des problèmes de conformité à l'aide d'EventBridge (p. 785).

Affichage des données de conformité actuelles (AWS CLI)

Vous pouvez afficher des récapitulatifs des données de conformité pour l'application de correctifs, les associations et les types de conformité personnalisée dans l'AWS CLI en utilisant les commandes d'AWS CLI suivantes.

list-compliance-summaries

Renvoie un décompte récapitulatif des statuts d'association conformes et non conformes en fonction du filtre que vous spécifiez. (API : ListComplianceSummaries)

list-resource-compliance-summaries

Renvoie un récapitulatif du nombre au niveau des ressources. Ce récapitulatif inclut des informations sur les statuts Conforme et Non conforme et les nombres détaillés de gravité de l'élément de conformité, en fonction des critères de filtre que vous spécifiez. (API : ListResourceComplianceSummaries)

Vous pouvez afficher des données de conformité supplémentaires pour l'application de correctifs à l'aide des commandes d'AWS CLI suivantes.

describe-patch-group-state

Renvoie l'état des informations globales de conformité des correctifs pour un groupe de correctifs. (API : DescribePatchGroupState)

describe-instance-patch-states-for-patch-group

Renvoie l'état des correctifs de haut niveau pour les instances du groupe de correctifs spécifié. (API : DescribeInstancePatchStatesForPatchGroup)

Note

Pour savoir comment configurer l'application des correctifs et afficher les détails de conformité des correctifs à l'aide de l'AWS CLI, consultez Procédure : Appliquer un correctif à un environnement serveur (AWS CLI) (p. 1168).

Affichage du suivi des modifications et de l'historique de configuration de la conformité

Le service Conformité de configuration Systems Manager affiche les données actuelles de conformité des associations et de l'application de correctifs pour vos instances gérées. Vous pouvez afficher le suivi des modifications et l'historique de conformité des correctifs et des associations à l'aide d'AWS Config. AWS Config fournit une vue détaillée de la configuration des ressources AWS dans votre compte AWS. Elle indique comment les ressources sont liées entre elles et comment elles ont été configurées dans le passé, pour que vous puissiez observer comment les configurations et les relations changent au fil du temps. Pour afficher le suivi des modifications et l'historique des associations et de l'application de correctifs, vous devez activer les ressources suivantes dans AWS Config :

- SSM:PatchCompliance
- SSM:AssociationCompliance

Pour de plus amples informations sur le choix et la configuration de ces ressources spécifiques dans AWS Config, veuillez consulter Sélection des ressources enregistrées par AWS Config dans le AWS Config Developer Guide.

Note

Pour plus d'informations sur la tarification AWS Config, consultez Tarification .

Correction des problèmes de conformité à l'aide d'EventBridge

Vous pouvez corriger rapidement les problèmes liés à la conformité des correctifs et des associations en utilisant Systems Manager Fonctionnalité Exécuter la commande . Vous pouvez cibler une instance

IDs ou Amazon EC2 des balises et exécuter le `AWS-RunPatchBaseline` document ou le `AWS-RefreshAssociation` document. Si actualiser l'association ou exécuter à nouveau la référence de correctif ne permet pas de résoudre le problème de conformité, vous devez vérifier vos associations, références de correctifs ou configurations d'instance pour comprendre pourquoi les exécutions de la fonctionnalité Fonctionnalité Exécuter la commande n'ont pas permis de résoudre le problème.

Pour de plus amples informations sur l'application de correctifs, veuillez consulter AWS Systems Manager Gestionnaire de correctifs (p. 1047) et À propos du Document SSM AWS-RunPatchBaseline (p. 1080).

Pour de plus amples informations sur les associations, veuillez consulter Utilisation d'associations dans Systems Manager (p. 989).

Pour de plus amples informations sur l'exécution d'une commande, veuillez consulter Exécuter des commandes avec Systems Manager Fonctionnalité Exécuter la commande (p. 946).

Spécification du service Conformité de configuration comme cible d'un EventBridge événement

Vous pouvez également configurer EventBridge pour effectuer une action en réponse à des événements de Conformité de configuration. Par exemple, si une ou plusieurs instances ne parviennent pas à installer des mises à jour de correctif critiques ou à exécuter une association qui installe un logiciel antivirus, vous pouvez configurer EventBridge pour exécuter le `AWS-RunPatchBaseline` document ou le `AWS-RefreshAssocation` document lorsque l'événement Conformité de configuration se produit.

Utilisez la procédure suivante afin de configurer Conformité de configuration en tant que cible d'un EventBridge événement.

Pour configurer Conformité de configuration comme cible d'un événement EventBridge (console)

1. Ouvrez la console Amazon EventBridge à l'adresse https://console.aws.amazon.com/events/.
2. Dans le volet de navigation, choisissez Rules (Règles), puis Create rule (Créer une règle).

   -ou-

   Si la page Amazon EventBridge d'accueil s'ouvre d'abord, choisissez Create rule (Créer une règle).
3. Saisissez un nom et une description pour la règle.

   Une règle ne peut pas avoir le même nom qu'une autre règle de la même région et sur le même bus d'événement.
4. Pour Define pattern (Définir un modèle), choisissez Event pattern (Modèle d'événement). Le modèle d'événement vous permet de créer une règle qui génère des événements pour des actions spécifiques dans les services AWS.
5. Choisissez Pré-defined pattern by service (Modèle prédéfini par service).
6. Pour Service provider (Fournisseur de services), choisissez AWS.
7. Pour Service Name (Nom du service), choisissez Systems Manager EC2 Simple (SSM)
8. Pour le type d'événement, choisissez Conformité de configuration.
9. Pour Select event bus (Sélectionner un bus d'événement), choisissez le bus d'événement que vous souhaitez associer à cette règle. Si vous souhaitez que cette règle se déclenche sur les événements correspondants provenant de votre propre compte AWS, sélectionnez AWS default event bus (Bus d'événement par défaut AWS). Lorsqu'un service AWS de votre compte émet un événement, il accède toujours au bus d'événement par défaut de votre compte.
10. Pour Target (Cible), choisissez SSM Run Command (Exécuter la commande SSM).
11. Dans la liste Document, choisissez un document SSM à exécuter lorsque la cible sera invoquée. Par exemple, choisissez `AWS-RunPatchBaseline` pour un événement de correctif non conforme ou choisissez `AWS-RefreshAssociation` pour un événement d'association non conforme.
12. Spécifiez les informations pour les champs et paramètres restants.

Note

Les champs et paramètres requis sont dotés d'une astérisque (*) en regard de leur nom. Pour créer une cible, vous devez spécifier une valeur pour chaque paramètre ou champ requis. Si vous ne le faites pas, le système crée la règle mais elle n'est pas exécutée.

13. (Facultatif) Entrez une ou plusieurs balises pour la règle. Pour de plus amples informations, veuillez consulter Balisage de vos ressources Amazon EventBridge dans le Guide de l'utilisateur Amazon EventBridge.
14. Choisissez Create et exécutez l'assistant.

Procédure pas à pas de Conformité de la configuration (AWS CLI)

La procédure suivante vous guide à travers le processus d'utilisation de l'action PutComplianceItems d'API pour attribuer des métadonnées de conformité personnalisées à une ressource. Vous pouvez aussi utiliser cette action d'API pour attribuer manuellement des métadonnées de conformité des correctifs ou des associations à une instance, comme indiqué dans la procédure pas à pas suivante. Pour de plus amples informations sur la conformité personnalisée, veuillez consulter A propos de la conformité personnalisée (p. 783).

Pour attribuer des métadonnées de conformité personnalisée à une instance gérée (AWS CLI)

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).
2. Exécutez la commande suivante pour attribuer des métadonnées de conformité personnalisées à une instance. Actuellement, le seul type de ressource pris en charge est `ManagedInstance`.

   Linux

   ```
   aws ssm put-compliance-items \
       --resource-id instance_ID \
       --resource-type ManagedInstance \
       --compliance-type Custom:user-defined_string \
       --execution-summary ExecutionTime=user-defined_time_and/or_date_value \
       --items Id=user-defined_ID,Title=user-defined_title,Severity=one_or_more_comma-
   separated_severities:CRITICAL, MAJOR, MINOR,INFORMATIONAL, or
    UNSPECIFIED,Status=COMPLIANT or NON_COMPLIANT
   ```

   Windows

   ```
   aws ssm put-compliance-items ^
       --resource-id instance_ID ^
       --resource-type ManagedInstance ^
       --compliance-type Custom:user-defined_string ^
       --execution-summary ExecutionTime=user-defined_time_and/or_date_value ^
       --items Id=user-defined_ID,Title=user-defined_title,Severity=one_or_more_comma-
   separated_severities:CRITICAL, MAJOR, MINOR,INFORMATIONAL, or
    UNSPECIFIED,Status=COMPLIANT or NON_COMPLIANT
   ```
3. Répétez l'étape précédente pour attribuer des métadonnées supplémentaires de conformité personnalisée à une ou plusieurs instances. Vous pouvez aussi utiliser les commandes suivantes pour attribuer manuellement des métadonnées de conformité des correctifs ou des associations à des instances gérées :

Métadonnées de conformité des associations

Linux

```
aws ssm put-compliance-items \
    --resource-id instance_ID \
    --resource-type ManagedInstance \
    --compliance-type Association \
    --execution-summary ExecutionTime=user-defined_time_and/or_date_value \
    --items Id=user-defined_ID,Title=user-defined_title,Severity=one_or_more_comma-
separated_severities:CRITICAL, MAJOR, MINOR,INFORMATIONAL, or
 UNSPECIFIED,Status=COMPLIANT or NON_COMPLIANT
```

Windows

```
aws ssm put-compliance-items ^
    --resource-id instance_ID ^
    --resource-type ManagedInstance ^
    --compliance-type Association ^
    --execution-summary ExecutionTime=user-defined_time_and/or_date_value ^
    --items Id=user-defined_ID,Title=user-defined_title,Severity=one_or_more_comma-
separated_severities:CRITICAL, MAJOR, MINOR,INFORMATIONAL, or
 UNSPECIFIED,Status=COMPLIANT or NON_COMPLIANT
```

Métadonnées de conformité des correctifs

Linux

```
aws ssm put-compliance-items \
    --resource-id instance_ID \
    --resource-type ManagedInstance \
    --compliance-type Patch \
    --execution-summary ExecutionTime=user-defined_time_and/
or_date_value,ExecutionId=user-defined_ID,ExecutionType=Command  \
    --items Id=for_example, KB12345,Title=user-
defined_title,Severity=one_or_more_comma-separated_severities:CRITICAL,
 MAJOR, MINOR,INFORMATIONAL, or UNSPECIFIED,Status=COMPLIANT or
 NON_COMPLIANT,Details="{PatchGroup=name_of_group,PatchSeverity=the_patch_severity,
 for example, CRITICAL}"
```

Windows

```
aws ssm put-compliance-items ^
    --resource-id instance_ID ^
    --resource-type ManagedInstance ^
    --compliance-type Patch ^
    --execution-summary ExecutionTime=user-defined_time_and/
or_date_value,ExecutionId=user-defined_ID,ExecutionType=Command  ^
    --items Id=for_example, KB12345,Title=user-
defined_title,Severity=one_or_more_comma-separated_severities:CRITICAL,
 MAJOR, MINOR,INFORMATIONAL, or UNSPECIFIED,Status=COMPLIANT or
 NON_COMPLIANT,Details="{PatchGroup=name_of_group,PatchSeverity=the_patch_severity,
 for example, CRITICAL}"
```

4. Exécutez la commande suivante pour afficher la liste des éléments de conformité pour une instance gérée spécifique. Utilisez des filtres pour explorer en détail des données de conformité spécifiques.

Linux

```
aws ssm list-compliance-items \
    --resource-ids instance_ID \
    --resource-types ManagedInstance \
    --filters one_or_more_filters
```

Windows

```
aws ssm list-compliance-items ^
    --resource-ids instance_ID ^
    --resource-types ManagedInstance ^
    --filters one_or_more_filters
```

Les exemples suivants vous montrent comment utiliser cette commande avec des filtres.

Linux

```
aws ssm list-compliance-items \
    --resource-ids i-02573cafcfEXAMPLE \
    --resource-type ManagedInstance \
    --filters Key=DocumentName,Values=AWS-RunPowerShellScript
 Key=Status,Values=NON_COMPLIANT,Type=NotEqual
 Key=Id,Values=cee20ae7-6388-488e-8be1-a88ccEXAMPLE Key=Severity,Values=UNSPECIFIED
```

Windows

```
aws ssm list-compliance-items ^
    --resource-ids i-02573cafcfEXAMPLE ^
    --resource-type ManagedInstance ^
    --filters Key=DocumentName,Values=AWS-RunPowerShellScript
 Key=Status,Values=NON_COMPLIANT,Type=NotEqual
 Key=Id,Values=cee20ae7-6388-488e-8be1-a88ccEXAMPLE Key=Severity,Values=UNSPECIFIED
```

Linux

```
aws ssm list-resource-compliance-summaries \
    --filters Key=OverallSeverity,Values=UNSPECIFIED
```

Windows

```
aws ssm list-resource-compliance-summaries ^
    --filters Key=OverallSeverity,Values=UNSPECIFIED
```

Linux

```
aws ssm list-resource-compliance-summaries \
    --filters Key=OverallSeverity,Values=UNSPECIFIED
 Key=ComplianceType,Values=Association Key=InstanceId,Values=i-02573cafcfEXAMPLE
```

Windows

```
aws ssm list-resource-compliance-summaries ^
    --filters Key=OverallSeverity,Values=UNSPECIFIED
 Key=ComplianceType,Values=Association Key=InstanceId,Values=i-02573cafcfEXAMPLE
```

5. Exécutez la commande suivante pour afficher un récapitulatif des statuts de conformité. Utilisez des filtres pour explorer en détail des données de conformité spécifiques.

```
aws ssm list-resource-compliance-summaries --filters One or more filters.
```

Les exemples suivants vous montrent comment utiliser cette commande avec des filtres.

Linux

```
aws ssm list-resource-compliance-summaries \
    --filters Key=ExecutionType,Values=Command
```

Windows

```
aws ssm list-resource-compliance-summaries ^
    --filters Key=ExecutionType,Values=Command
```

Linux

```
aws ssm list-resource-compliance-summaries \
    --filters Key=AWS:InstanceInformation.PlatformType,Values=Windows
 Key=OverallSeverity,Values=CRITICAL
```

Windows

```
aws ssm list-resource-compliance-summaries ^
    --filters Key=AWS:InstanceInformation.PlatformType,Values=Windows
 Key=OverallSeverity,Values=CRITICAL
```

6. Exécutez la commande suivante pour afficher un récapitulatif du nombre de ressources conformes et non conformes pour un type de conformité. Utilisez des filtres pour explorer en détail des données de conformité spécifiques.

```
aws ssm list-compliance-summaries --filters One or more filters.
```

Les exemples suivants vous montrent comment utiliser cette commande avec des filtres.

Linux

```
aws ssm list-compliance-summaries \
    --filters Key=AWS:InstanceInformation.PlatformType,Values=Windows
 Key=PatchGroup,Values=TestGroup
```

Windows

```
aws ssm list-compliance-summaries ^
```

```
    --filters Key=AWS:InstanceInformation.PlatformType,Values=Windows
 Key=PatchGroup,Values=TestGroup
```

Linux

```
aws ssm list-compliance-summaries \
    --filters Key=AWS:InstanceInformation.PlatformType,Values=Windows
 Key=ExecutionId,Values=4adf0526-6aed-4694-97a5-14522EXAMPLE
```

Windows

```
aws ssm list-compliance-summaries ^
    --filters Key=AWS:InstanceInformation.PlatformType,Values=Windows
 Key=ExecutionId,Values=4adf0526-6aed-4694-97a5-14522EXAMPLE
```

Inventaire AWS Systems Manager

L'inventaire AWS Systems Manager fournit une meilleure visibilité sur votre environnement informatique Amazon EC2 et sur site. Vous pouvez utiliser l'inventaire pour collecter les métadonnées à partir de vos instances gérées. Vous pouvez stocker ces métadonnées dans un compartiment Amazon Simple Storage Service (Amazon S3) central, puis utiliser les outils intégrés pour interroger les données et déterminer rapidement quelles instances exécutent les logiciels et les configurations requis par votre stratégie logicielle, et quelles instances doivent être mises à jour. Vous pouvez configurer l'inventaire sur l'ensemble de vos instances gérées à l'aide d'une procédure en un clic. Vous pouvez également configurer et afficher les données d'inventaire de plusieurs régions et comptes AWS.

Si les types de métadonnées préconfigurés collectés par l'inventaire Systems Manager ne répondent pas à vos besoins, vous pouvez créer un inventaire personnalisé. L'inventaire personnalisé est simplement un fichier JSON contenant des informations que vous fournissez et ajoutez à l'instance gérée dans un répertoire spécifique. Lorsque l'inventaire Systems Manager collecte les données, il capture les données de cet inventaire personnalisé. Par exemple, si vous exécutez un centre de données volumineux, vous pouvez spécifier l'emplacement des racks de chacun de vos serveurs en tant qu'inventaire personnalisé. Vous pouvez ensuite consulter les données de l'espace rack lorsque vous consultez les autres données d'inventaire.

> **Important**
>
> L'inventaire Systems Manager collecte uniquement les métadonnées de vos instances gérées. L'inventaire n'accède pas aux données ou aux informations propriétaires.

Le tableau suivant répertorie les types de métadonnées que vous pouvez collecter avec l'inventaire Systems Manager. Ce tableau répertorie également les instances depuis lesquelles vous pouvez collecter des informations d'inventaire et les intervalles de collecte que vous pouvez spécifier.

Configuration	Details (Détails)
Types de métadonnées	Vous pouvez configurer l'inventaire pour collecter les types de métadonnées suivants : • Applications : Noms d'application, éditeurs, versions, etc. • Composants AWS : Pilote EC2, agents, versions, etc.

Configuration	Details (Détails)
	• Fichiers : Nom, taille, version, date d'installation, heures de modification et des derniers accès, etc. • Configuration réseau : Adresse IP, adresse MAC, DNS, passerelle, masque de sous-réseau, etc. • Mises à jour Windows : ID du correctif, installé par, date d'installation, etc. • Détails de l'instance : Nom du système, nom des systèmes d'exploitation, version du système d'exploitation, dernier démarrage, DNS, domaine, groupe de travail, architecture du système d'exploitation, etc. • Services : Nom, nom d'affichage, statut, services dépendants, type de service, type de démarrage, etc. • Balises : Balises attribuées à vos instances. • Registre Windows : Chemin de la clé de registre, nom de valeur, type de valeur et valeur. • Rôles Windows : Nom, nom d'affichage, chemin, type de fonction, état installé, etc. • Inventaire personnalisé : Métadonnées affectées à une instance gérée comme décrit dans Utilisation de l'inventaire personnalisé (p. 826) . Note Pour afficher la liste de toutes les métadonnées collectées par l'inventaire, consultez Métadonnées collectées par inventaire (p. 794).
Instances à partir desquelles collecter les informations	Vous pouvez choisir d'inventorier toutes les instances dans votre compte AWS, ou de sélectionner individuellement les instances ou les groupes cibles d'instances, en utilisant des balises Amazon EC2. Pour plus d'informations sur l'exécution de la collecte d'inventaire sur l'ensemble de vos instances, consultez Effectuer l'inventaire de toutes les instances gérées dans votre compte AWS (p. 807).
Quand collecter les informations	Vous pouvez spécifier un intervalle de collecte en minutes, heures, jours et semaines. L'intervalle de collecte le plus court est toutes les 30 minutes.

Note

En fonction de la quantité de données recueillies, le système peut prendre plusieurs minutes pour présenter les données à la sortie que vous avez spécifiée. Une fois que les informations ont été

collectées, les métadonnées sont envoyées via un canal HTTPS sécurisé à un magasin AWS en texte brut qui n'est accessible qu'à partir de votre compte AWS.

Vous pouvez afficher les données dans la console AWS Systems Manager sur la page Inventory (Inventaire), qui comprend plusieurs cartes prédéfinies pour vous aider à interroger les données.

Note

Les cartes d'inventaire filtrent automatiquement les instances gérées Amazon EC2 ayant l'état Terminated (Résilié) et Stopped (Arrêté). Pour les instances gérées sur site, les cartes d'inventaire filtrent automatiquement les instances ayant l'état Terminated (Résilié).

Si vous créez une synchronisation de données de ressources pour synchroniser et stocker toutes vos données dans un seul compartiment S3, vous pouvez explorer en détail les données sur la page Inventory Detailed View (Vue détaillée d'inventaire). Pour plus d'informations, consultez la section Interrogation des données d'inventaire à partir de plusieurs régions et comptes (p. 811).

Prise en charge de Amazon EventBridge

Cette fonction Systems Manager est prise en charge en tant que type d'événement dans les règles EventBridge. Pour plus d'informations, consultez Surveillance des événements Systems Manager avec

Amazon EventBridge (p. 1340) et Référence : modèles et types Amazon EventBridge d'événements pourSystems Manager (p. 1387).

Table des matières

- En savoir plus sur l'inventaire Systems Manager (p. 794)
- Configuration de la synchronisation de données de ressource pour l'inventaire (p. 800)
- Configuration de la collecte d'inventaire (p. 806)
- Utilisation des données d'inventaire Systems Manager (p. 810)
- Utilisation de l'inventaire personnalisé (p. 826)
- Affichage de l'historique d'inventaire et suivi des modifications (p. 836)
- Procédures pas à pas de l'inventaire Systems Manager (p. 837)
- Dépannage des problèmes liés à l'inventaire Systems Manager (p. 849)

En savoir plus sur l'inventaire Systems Manager

Lorsque vous configurez l'inventaire Systems Manager, vous spécifiez le type de métadonnées à collecter, les instances à partir desquelles les métadonnées doivent être collectées et un calendrier de collecte des métadonnées. Ces configurations sont enregistrées avec votre compte AWS sous la forme d'une association Systems Manager State Manager. Une association est simplement une configuration.

> **Note**
>
> L'inventaire collecte uniquement les métadonnées. Il ne recueille aucune donnée personnelle ou propriétaire.

Rubriques

- Métadonnées collectées par inventaire (p. 794)
- Utilisation de l'inventaire de fichiers et du registre Windows (p. 798)
- Services AWS connexes (p. 799)

Métadonnées collectées par inventaire

L'exemple suivant affiche la liste complète des métadonnées collectées par chaque plug-in Inventory.

```
[
  {
    "typeName": "AWS:InstanceInformation",
    "version": "1.0",
    "attributes":[
      { "name": "AgentType",                                "dataType" : "STRING"},
      { "name": "AgentVersion",                             "dataType" : "STRING"},
      { "name": "ComputerName",                             "dataType" : "STRING"},
      { "name": "IamRole",                                  "dataType" : "STRING"},
      { "name": "InstanceId",                               "dataType" : "STRING"},
      { "name": "IpAddress",                                "dataType" : "STRING"},
      { "name": "PlatformName",                             "dataType" : "STRING"},
      { "name": "PlatformType",                             "dataType" : "STRING"},
      { "name": "PlatformVersion",                          "dataType" : "STRING"},
      { "name": "ResourceType",                             "dataType" : "STRING"}
    ]
  },
  {
    "typeName" : "AWS:Application",
    "version": "1.1",
```

```
    "attributes":[
      { "name": "Name",                "dataType": "STRING"},
      { "name": "ApplicationType",     "dataType": "STRING"},
      { "name": "Publisher",           "dataType": "STRING"},
      { "name": "Version",             "dataType": "STRING"},
      { "name": "InstalledTime",       "dataType": "STRING"},
      { "name": "Architecture",        "dataType": "STRING"},
      { "name": "URL",                 "dataType": "STRING"},
      { "name": "Summary",             "dataType": "STRING"},
      { "name": "PackageId",           "dataType": "STRING"},
      { "name": "Release",             "dataType": "STRING"},
      { "name": "Epoch",               "dataType": "STRING"}
    ]
  },
  {
    "typeName" : "AWS:File",
    "version": "1.0",
    "attributes":[
      { "name": "Name",                "dataType": "STRING"},
      { "name": "Size",                "dataType": "STRING"},
      { "name": "Description",         "dataType": "STRING"},
      { "name": "FileVersion",         "dataType": "STRING"},
      { "name": "InstalledDate",       "dataType": "STRING"},
      { "name": "ModificationTime",    "dataType": "STRING"},
      { "name": "LastAccessTime",      "dataType": "STRING"},
      { "name": "ProductName",         "dataType": "STRING"},
      { "name": "InstalledDir",        "dataType": "STRING"},
      { "name": "ProductLanguage",     "dataType": "STRING"},
      { "name": "CompanyName",         "dataType": "STRING"},
      { "name": "ProductVersion",      "dataType": "STRING"}
    ]
  },
  {
    "typeName": "AWS:AWSComponent",
    "version": "1.0",
    "attributes":[
      { "name": "Name",                "dataType": "STRING"},
      { "name": "ApplicationType",     "dataType": "STRING"},
      { "name": "Publisher",           "dataType": "STRING"},
      { "name": "Version",             "dataType": "STRING"},
      { "name": "InstalledTime",       "dataType": "STRING"},
      { "name": "Architecture",        "dataType": "STRING"},
      { "name": "URL",                 "dataType": "STRING"}
    ]
  },
  {
    "typeName": "AWS:WindowsUpdate",
    "version":"1.0",
    "attributes":[
      { "name": "HotFixId",            "dataType": "STRING"},
      { "name": "Description",         "dataType": "STRING"},
      { "name": "InstalledTime",       "dataType": "STRING"},
      { "name": "InstalledBy",         "dataType": "STRING"}
    ]
  },
  {
    "typeName": "AWS:Network",
    "version":"1.0",
    "attributes":[
      { "name": "Name",                "dataType": "STRING"},
      { "name": "SubnetMask",          "dataType": "STRING"},
      { "name": "Gateway",             "dataType": "STRING"},
      { "name": "DHCPServer",          "dataType": "STRING"},
      { "name": "DNSServer",           "dataType": "STRING"},
      { "name": "MacAddress",          "dataType": "STRING"},
      { "name": "IPV4",                "dataType": "STRING"},
```

```
      { "name": "IPV6",               "dataType": "STRING"}
    ]
  },
  {
    "typeName": "AWS:PatchSummary",
    "version":"1.0",
    "attributes":[
      { "name": "PatchGroup",                   "dataType": "STRING"},
      { "name": "BaselineId",                    "dataType": "STRING"},
      { "name": "SnapshotId",                    "dataType": "STRING"},
      { "name": "OwnerInformation",              "dataType": "STRING"},
      { "name": "InstalledCount",                "dataType": "NUMBER"},
      { "name": "InstalledOtherCount",           "dataType": "NUMBER"},
      { "name": "NotApplicableCount",            "dataType": "NUMBER"},
      { "name": "MissingCount",                  "dataType": "NUMBER"},
      { "name": "FailedCount",                   "dataType": "NUMBER"},
      { "name": "OperationType",                 "dataType": "STRING"},
      { "name": "OperationStartTime",            "dataType": "STRING"},
      { "name": "OperationEndTime",              "dataType": "STRING"}
    ]
  },
  {
    "typeName": "AWS:ComplianceItem",
    "version":"1.0",
    "attributes":[
      { "name": "ComplianceType",                "dataType": "STRING"},
      { "name": "ExecutionId",                   "dataType": "STRING"},
      { "name": "ExecutionType",                 "dataType": "STRING"},
      { "name": "ExecutionTime",                 "dataType": "STRING"},
      { "name": "Id",                            "dataType": "STRING"},
      { "name": "Title",                         "dataType": "STRING"},
      { "name": "Status",                        "dataType": "STRING"},
      { "name": "Severity",                      "dataType": "STRING"},
      { "name": "DocumentName",                  "dataType": "STRING"},
      { "name": "DocumentVersion",               "dataType": "STRING"},
      { "name": "Classification",                "dataType": "STRING"},
      { "name": "PatchBaselineId",               "dataType": "STRING"},
      { "name": "PatchSeverity",                 "dataType": "STRING"},
      { "name": "PatchState",                    "dataType": "STRING"},
      { "name": "PatchGroup",                    "dataType": "STRING"},
      { "name": "InstalledTime",                 "dataType": "STRING"}
    ]
  },
  {
    "typeName": "AWS:ComplianceSummary",
    "version":"1.0",
    "attributes":[
      { "name": "ComplianceType",                  "dataType": "STRING"},
      { "name": "PatchGroup",                      "dataType": "STRING"},
      { "name": "PatchBaselineId",                 "dataType": "STRING"},
      { "name": "Status",                          "dataType": "STRING"},
      { "name": "OverallSeverity",                 "dataType": "STRING"},
      { "name": "ExecutionId",                     "dataType": "STRING"},
      { "name": "ExecutionType",                   "dataType": "STRING"},
      { "name": "ExecutionTime",                   "dataType": "STRING"},
      { "name": "CompliantCriticalCount",          "dataType": "NUMBER"},
      { "name": "CompliantHighCount",              "dataType": "NUMBER"},
      { "name": "CompliantMediumCount",            "dataType": "NUMBER"},
      { "name": "CompliantLowCount",               "dataType": "NUMBER"},
      { "name": "CompliantInformationalCount",     "dataType": "NUMBER"},
      { "name": "CompliantUnspecifiedCount",       "dataType": "NUMBER"},
      { "name": "NonCompliantCriticalCount",       "dataType": "NUMBER"},
      { "name": "NonCompliantHighCount",           "dataType": "NUMBER"},
      { "name": "NonCompliantMediumCount",         "dataType": "NUMBER"},
      { "name": "NonCompliantLowCount",            "dataType": "NUMBER"},
      { "name": "NonCompliantInformationalCount", "dataType": "NUMBER"},
```

```
      { "name": "NonCompliantUnspecifiedCount",   "dataType": "NUMBER"}
    ]
  },
  {
    "typeName": "AWS:InstanceDetailedInformation",
    "version":"1.0",
    "attributes":[
      { "name": "CPUModel",                     "dataType": "STRING"},
      { "name": "CPUCores",                     "dataType": "NUMBER"},
      { "name": "CPUs",                         "dataType": "NUMBER"},
      { "name": "CPUSpeedMHz",                  "dataType": "NUMBER"},
      { "name": "CPUSockets",                   "dataType": "NUMBER"},
      { "name": "CPUHyperThreadEnabled",        "dataType": "STRING"},
      { "name": "OSServicePack",                "dataType": "STRING"}
    ]
   },
   {
     "typeName": "AWS:Service",
     "version":"1.0",
     "attributes":[
       { "name": "Name",                        "dataType": "STRING"},
       { "name": "DisplayName",                 "dataType": "STRING"},
       { "name": "ServiceType",                 "dataType": "STRING"},
       { "name": "Status",                      "dataType": "STRING"},
       { "name": "DependentServices",            "dataType": "STRING"},
       { "name": "ServicesDependedOn",           "dataType": "STRING"},
       { "name": "StartType",                   "dataType": "STRING"}
     ]
    },
    {
      "typeName": "AWS:WindowsRegistry",
      "version":"1.0",
      "attributes":[
        { "name": "KeyPath",                        "dataType": "STRING"},
        { "name": "ValueName",                      "dataType": "STRING"},
        { "name": "ValueType",                      "dataType": "STRING"},
        { "name": "Value",                          "dataType": "STRING"}
      ]
    },
    {
      "typeName": "AWS:WindowsRole",
      "version":"1.0",
      "attributes":[
        { "name": "Name",                        "dataType": "STRING"},
        { "name": "DisplayName",                 "dataType": "STRING"},
        { "name": "Path",                        "dataType": "STRING"},
        { "name": "FeatureType",                 "dataType": "STRING"},
        { "name": "DependsOn",                   "dataType": "STRING"},
        { "name": "Description",                 "dataType": "STRING"},
        { "name": "Installed",                   "dataType": "STRING"},
        { "name": "InstalledState",              "dataType": "STRING"},
        { "name": "SubFeatures",                 "dataType": "STRING"},
        { "name": "ServerComponentDescriptor",   "dataType": "STRING"},
        { "name": "Parent",                      "dataType": "STRING"}
      ]
    }
]
```

Note

Avec le lancement de la version 2.5, le gestionnaire de package RPM a remplacé l'attribut Serial par Epoch. L'attribut Epoch est un entier qui augmente de façon monotone, comme Serial. Lorsque vous procédez à l'inventaire à l'aide du type AWS:Application, notez qu'une valeur supérieure pour Epoch désigne une version plus récente. Si les valeurs Epoch sont les mêmes

ou sont vides, utilisez la valeur de l'attribut Version ou Release (publication) pour déterminer la version plus récente.

Utilisation de l'inventaire de fichiers et du registre Windows

L'inventaire Systems Manager vous permet de rechercher et d'inventorier des fichiers sur les systèmes d'exploitation Windows et Linux. Vous pouvez également rechercher et inventorier le registre Windows.

Fichiers : Vous pouvez collecter des informations de métadonnées sur les fichiers, notamment leurs noms, l'heure de leur création, l'heure de leur dernière modification et de leur dernier accès, leur taille, etc. Pour commencer la collecte d'un inventaire de fichiers, indiquez le chemin d'accès où effectuer l'inventaire, un ou plusieurs modèles définissant les types de fichiers à inventorier, et si le chemin doit être parcouru de manière récursive. Systems Manager inventorie toutes les métadonnées des fichiers qui figurent dans le chemin indiqué correspondant au modèle. L'inventaire de fichiers utilise l'entrée de paramètre suivante.

```
{
"Path": string,
"Pattern": array[string],
"Recursive": true,
"DirScanLimit" : number // Optional
}
```

- Chemin : Chemin du répertoire où vous voulez inventorier les fichiers. Sous Windows, vous pouvez utiliser des variables d'environnement telles que %PROGRAMFILES% dès lors que la variable est mappée à un seul chemin d'accès au répertoire. Par exemple, si vous utilisez la variable %PATH% qui est mappée sur plusieurs chemins de répertoire, l'inventaire renvoie une erreur.
- Modèle : Tableau de modèles pour identifier les fichiers.
- Récursif : Valeur booléenne indiquant si l'inventaire doit parcourir de manière récursive les répertoires.
- DirScanLimit: Valeur facultative spécifiant le nombre de répertoires à analyser. Utilisez ce paramètre pour minimiser l'impact sur les performances de vos instances. Par défaut, l'inventaire analyse 5 000 répertoires au maximum.

  Note

  L'inventaire collecte des métadonnées pour 500 fichiers au maximum sur tous les chemins indiqués.

Voici des exemples de spécification de paramètres lors de l'exécution d'un inventaire de fichiers.

- Sous Linux, collectez les métadonnées des fichiers .sh dans le répertoire `/home/ec2-user`, en excluant tous les sous-répertoires.

```
[{"Path":"/home/ec2-user","Pattern":["*.sh", "*.sh"],"Recursive":false}]
```

- Sous Windows, collectez les métadonnées de tous les fichiers .exe du dossier Program Files, en incluant les sous-répertoires de manière récursive.

```
[{"Path":"C:\Program Files","Pattern":["*.exe"],"Recursive":true}]
```

- Sous Windows, collectez les métadonnées de modèles de journaux spécifiques.

```
[{"Path":"C:\ProgramData\Amazon","Pattern":["*amazon*.log"],"Recursive":true}]
```

- Limitez le nombre de répertoires lors de l'exécution d'une collection récursive.

```
[{"Path":"C:\Users","Pattern":["*.ps1"],"Recursive":true, "DirScanLimit": 1000}]
```

Registre Windows : Vous pouvez collecter des clés et valeurs de registre Windows. Vous pouvez choisir un chemin de clé et collecter toutes les clés et valeurs de manière récursive. Vous pouvez également collecter une clé de registre spécifique et sa valeur pour un chemin donné. Inventory collecte le chemin de clé, le nom, le type et la valeur.

```
{
"Path": string,
"Recursive": true,
"ValueNames": array[string] // optional
}
```

- Chemin : Chemin d'accès à la clé de registre.
- Récursif : Valeur booléenne indiquant si l'inventaire doit parcourir de manière récursive les chemins d'accès du registre.
- ValueNames: Tableau de noms de valeurs pour effectuer l'inventaire des clés de registre. Si vous utilisez ce paramètre, Systems Manager n'inventorie que les noms de valeurs spécifiés pour le chemin indiqué.

> **Note**
>
> L'inventaire collecte 250 valeurs de clé de registre au maximum sur tous les chemins indiqués.

Voici des exemples de spécification de paramètres lors de l'exécution d'un inventaire du registre Windows.

- Collectez toutes les clés et valeurs de manière récursive pour un chemin spécifique.

```
[{"Path":"HKEY_LOCAL_MACHINE\SOFTWARE\Amazon","Recursive": true}]
```

- Collectez toutes les clés et valeurs pour un chemin spécifique (recherche récursive désactivée).

```
[{"Path":"HKEY_LOCAL_MACHINE\SOFTWARE\Intel\PSIS\PSIS_DECODER", "Recursive": false}]
```

- Collectez une clé spécifique à l'aide de l'option `ValueNames`.

```
{"Path":"HKEY_LOCAL_MACHINE\SOFTWARE\Amazon\MachineImage","ValueNames":["AMIName"]}
```

Services AWS connexes

L'inventaire Systems Manager fournit un instantané de votre inventaire actuel pour vous aider à gérer la stratégie logicielle et à améliorer la sécurité de votre parc d'instances complet. Vous pouvez étendre vos capacités de gestion d'inventaire et de migration avec les services AWS suivants.

- AWS Config fournit un historique des modifications apportées à votre inventaire, ainsi que la possibilité de créer des règles pour générer des notifications lorsqu'un élément de configuration est modifié. Pour de plus amples informations, veuillez consulter Enregistrement d'un inventaire d'instances gérées Amazon EC2 dans le AWS Config Developer Guide.
- AWS Application Discovery Service est conçu pour collecter l'inventaire en fonction du type de système d'exploitation, de l'inventaire des applications, des processus, des connexions et des métriques de performances du serveur à partir de votre système d'exploitation sur site VMs afin de prendre en charge une migration réussie vers AWS. Pour de plus amples informations, veuillez consulter Application Discovery Service Guide de l'utilisateur.

Configuration de la synchronisation de données de ressource pour l'inventaire

Cette rubrique décrit comment configurer la synchronisation des données de ressource pour l'inventaire Systems Manager. Pour de plus amples informations sur la synchronisation des données de ressource pour Systems Manager Explorer, veuillez consulter Configuration de Systems Manager Explorer de sorte à afficher les données de plusieurs comptes et régions (p. 163).

A propos de la synchronisation des données de ressource

Vous pouvez utiliser la synchronisation de données de ressource Systems Manager pour envoyer les données d'inventaire collectées à partir de toutes vos instances gérées vers un même compartiment S3. La synchronisation des données de ressource met alors automatiquement à jour les données centralisées lors de la collecte de nouvelles données d'inventaire. Avec toutes les données d'inventaire stockées dans un compartiment S3 cible, vous pouvez utiliser les services comme Amazon Athena et Amazon QuickSight pour questionner et analyser les données regroupées.

Par exemple, vous avez configure l'inventaire pour la collecte des données relatives au système d'exploitation (OS) et aux applications qui s'exécutent sur une série de 150 instances gérées. Certaines de ces instances sont localisées sur un centre de données hybride, d'autres sont exécutés dans Amazon EC2 parmi plusieurs régions AWS. Si vous n'avez pas configuré la synchronisation des données de ressource, vous devez soit rassembler manuellement les données d'inventaire collectées pour chaque instance, soit créer des scripts pour rassembler ces informations. Vous devriez alors transférer les données vers une application afin de pouvoir les interroger et les analyser.

Grâce à la synchronisation des données de ressource, vous pouvez synchroniser toutes les données d'inventaire provenant de toutes vos instances gérées en une seule opération. Une fois la synchronisation réalisée avec succès, Systems Manager crée une référence pour toutes les données d'inventaire et les enregistre dans le compartiment S3 cible. Une fois les nouvelles données d'inventaire collectées, Systems Manager met automatiquement à jour les données dans le compartiment S3. Vous pouvez alors transférer de manière rapide et économe les données vers Amazon Athena et Amazon QuickSight.

Le diagramme 1 montre comment la synchronisation de données de ressources rassemble les données d'inventaire provenant des instances gérées dans Amazon EC2 et d'un environnement hybride vers un compartiment S3 cible. Ce diagramme montre également comment la synchronisation des données de ressources fonctionne avec plusieurs comptes AWS et régions AWS.

Diagramme 1 : Synchronisation des données de ressources avec plusieurs comptes et régions AWS

Si vous supprimez une instance gérée, la synchronisation des données de ressource conserve le fichier d'inventaire pour l'instance supprimée. Néanmoins, pour les instances en cours d'exécution, la synchronisation des données de ressource écrase les anciens fichiers d'inventaire lors de la création et de l'écriture de nouveaux fichiers sur le compartiment S3. Si vous souhaitez suivre les modifications d'inventaire au fil du temps, vous pouvez utiliser le service AWS Config pour suivre le type de ressource `SSM:ManagedInstanceInventory`. Pour de plus amples informations, veuillez consulter Démarrer avec AWS Config.

Utilisez les procédures décrites dans cette section pour créer une synchronisation des données de ressource pour l'inventaire à l'aide des consoles Amazon S3 et AWS Systems Manager. Vous pouvez également utiliser AWS CloudFormation pour créer ou supprimer une synchronisation de données de ressources. Pour utiliser AWS CloudFormation, ajoutez la ressource AWS::SSM::ResourceDataSync à votre modèle AWS CloudFormation. Pour de plus amples informations, veuillez consulter l'une des ressources de documentation suivantes :

- Ressource CloudFormation AWS pour la synchronisation des données de ressource dans AWS Systems Manager (blog)
- Utilisation des modèles AWS CloudFormation dans le AWS CloudFormation Guide de l'utilisateur

> **Note**
>
> Vous pouvez utiliser AWS Key Management Service (AWS KMS) pour chiffrer les données d'inventaire dans le compartiment S3. Pour obtenir un exemple de création d'une synchronisation chiffrée en utilisant l'AWS CLI et d'utilisation des données centralisées dans Amazon Athena et Amazon QuickSight, consultez Procédure : Utiliser la synchronisation de données de ressources pour regrouper les données d'inventaire (p. 843).

Avant de commencer

Avant de créer une synchronisation des données de ressource, appliquez la procédure suivante pour créer un compartiment S3 central pour stocker les données d'inventaire agrégées. La procédure décrit comment affecter une stratégie de compartiment qui permet à Systems Manager d'écrire des données d'inventaire dans le compartiment à partir de plusieurs comptes. Si vous disposez déjà d'un compartiment S3 que vous

souhaitez utiliser pour agréger les données d'inventaire pour la synchronisation des données de ressource, vous devez configurer celui-ci pour qu'il utilise la stratégie dans la procédure suivante.

Pour créer et configurer un compartiment S3 pour la synchronisation de données de ressources

1. Ouvrez la console Amazon S3 à l'adresse https://console.aws.amazon.com/s3/.
2. Créez un compartiment pour stocker vos données d'inventaire rassemblées. Pour de plus amples informations, veuillez consulter Créer un compartiment dans le Amazon Simple Storage Service Guide de mise en route. Notez le nom du compartiment et la région AWS dans laquelle vous l'avez créé.
3. Sélectionnez l'onglet Autorisations, puis Stratégie de compartiment.
4. Copiez et collez la stratégie de compartiment suivante dans l'éditeur de stratégie. Remplacez *DOC-EXAMPLE-BUCKET* et *account-id* par le nom du compartiment S3 que vous avez créé et un ID de compte AWS valide.

   Pour activer plusieurs comptes AWS pour envoyer les données d'inventaire vers le compartiment Amazon S3 central, spécifiez chaque compte dans la stratégie, comme illustré dans l'exemple `Resource` suivant :

```
"Resource": [
     "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*/accountid=123456789012/*",
     "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*/accountid=444455556666/*",
     "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*/accountid=777788889999/*"
                ],
```

   (Facultatif) Remplacez *bucket-prefix* par le nom d'un Amazon S3 préfixe (sous-répertoire). Si vous n'avez pas créé de préfixe, supprimez *bucket-prefix/* depuis l'ARN dans la stratégie suivante.

   > **Note**
   >
   > Pour plus d'informations sur l'affichage de votre ID de compte AWS, consultez Votre ID de compte AWS et son alias dans le IAM Guide de l'utilisateur.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"SSMBucketPermissionsCheck",
         "Effect":"Allow",
         "Principal":{
            "Service":"ssm.amazonaws.com"
         },
         "Action":"s3:GetBucketAcl",
         "Resource":"arn:aws:s3:::DOC-EXAMPLE-BUCKET"
      },
      {
         "Sid":" SSMBucketDelivery",
         "Effect":"Allow",
         "Principal":{
            "Service":"ssm.amazonaws.com"
         },
         "Action":"s3:PutObject",
         "Resource":[
            "arn:aws:s3:::DOC-EXAMPLE-BUCKET/bucket-prefix/*/accountid=account-id-1/*",
            "arn:aws:s3:::DOC-EXAMPLE-BUCKET/bucket-prefix/*/accountid=account-id-2/*",
            "arn:aws:s3:::DOC-EXAMPLE-BUCKET/bucket-prefix/*/accountid=account-id-3/*"
         ],
         "Condition":{
            "StringEquals":{
               "s3:x-amz-acl":"bucket-owner-full-control"
            }
         }
```

```
        }
    ]
}
```

Note

La région Asie-Pacifique a été mise en ligne le 25 avril 2019. Si vous créez une synchronisation de données de ressources pour une région AWS qui est entrée en ligne depuis la région Asie-Pacifique (Hong Kong) (ap-east-1) ou une version ultérieure, vous devez saisir une entrée principale de service spécifique à la région dans la section `SSMBucketDelivery`. L'exemple suivant inclut une entrée principale de service spécifique à la région pour `ssm.ap-east-1.amazonaws.com`.

```
{
          "Sid":" SSMBucketDelivery",
          "Effect":"Allow",
          "Principal":{
             "Service":["ssm.amazonaws.com","ssm.ap-east-1.amazonaws.com"]
          },
```

Créer une synchronisation de données de ressources pour l'inventaire

Utilisez la procédure suivante pour créer une synchronisation de données de ressource pour l'inventaire Systems Manager avec la console Systems Manager. Pour plus d'informations sur la création d'une synchronisation de données de ressource à l'aide de l'AWS CLI, consultez Procédure : Configurer vos instances gérées pour l'inventaire à l'aide de l'interface de ligne de commande (p. 839).

Pour créer une synchronisation de données de ressources

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Instances gérées.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Instances gérées.
3. Choisissez Configure Inventory (Configurer l'inventaire), Resource Data Syncs (Synchronisations de données de ressources), puis Create resource data sync (Créer une synchronisation des données de ressource).
4. Dans le champ Nom de la synchronisation, entrez un nom pour la configuration de la synchronisation.
5. Dans le champ Nom du compartiment, entrez le nom du compartiment Amazon S3 créé au début de cette procédure.
6. (Facultatif) Dans le champ Préfixe du compartiment, tapez le nom d'un préfixe de compartiment S3 (sous-répertoire).
7. Dans le champ Région du compartiment, choisissez Cette région si le compartiment S3 créé est localisé dans la région AWS actuelle. Si le compartiment est localisé dans une autre région AWS, choisissez Autre région, et tapez le nom de la région.

   Note

   Si la synchronisation et le compartiment S3 cible sont localisés dans des régions différentes, vous pourriez être sujet à une tarification de transfert de données. Pour de plus amples informations, veuillez consulter Tarification Amazon S3.

8. (Facultatif) Dans le champ ARN de clé KMS, entrez ou collez un ARN de clé KMS pour chiffrer les données d'inventaire dans Amazon S3.
9. Sélectionnez Create.

Pour synchroniser les données d'inventaire issues de plusieurs régions AWS, vous devez créer une synchronisation de données de ressources dans chaque région. Répétez cette procédure dans chaque région AWS où vous souhaitez collecter des données d'inventaire et envoyez-les vers le compartiment S3 central. Lorsque vous créez la synchronisation dans chaque région, spécifiez le compartiment Amazon S3 central dans le champ Nom du compartiment. Ensuite, utilisez l'option Région d'un compartiment pour choisir la région où vous avez créé le compartiment Amazon S3 central, comme illustré dans la capture d'écran suivante. La prochaine fois que l'association s'exécute pour collecter les données d'inventaire, Systems Manager stocke les données dans le compartiment S3 central.

Création d'une synchronisation des données de ressource d'inventaire pour les comptes définis dansAWS Organizations

Vous pouvez synchroniser les données d'inventaire des comptes AWS définis dans AWS Organizations vers un Amazon S3 compartiment central. Une fois que vous avez terminé les procédures suivantes, les données d'inventaire sont synchronisées avec les préfixes de Amazon S3 clé individuels dans le compartiment central. Chaque préfixe de clé représente un ID de compte AWS différent.

Avant de commencer

Avant de commencer, vérifiez que vous avez installé et configuré des comptes AWS dans AWS Organizations . Pour de plus amples informations, consultez dans le Manuel de l'utilisateur AWS Organizations.

Sachez également que vous devez créer la synchronisation des données de ressource basée sur l'organisation pour chaque région AWS et chaque compte défini dans AWS Organizations .

Création d'un compartiment S3 central

Utilisez la procédure suivante pour créer un compartiment S3 central pour stocker les données d'inventaire agrégées. La procédure décrit comment affecter une stratégie de compartiment qui permet à Systems Manager d'écrire des données d'inventaire dans le compartiment à partir de ID de compte AWS

Organizations. Si vous disposez déjà d'un compartiment S3 que vous souhaitez utiliser pour agréger les données d'inventaire pour la synchronisation des données de ressource, vous devez configurer celui-ci pour qu'il utilise la stratégie dans la procédure suivante.

Pour créer et configurer un compartiment S3 pour la synchronisation de données de ressource pour plusieurs comptes définis dans AWS Organizations

1. Ouvrez la console Amazon S3 à l'adresse https://console.aws.amazon.com/s3/.
2. Créez un compartiment pour stocker vos données d'inventaire agrégées. Pour de plus amples informations, veuillez consulter Créer un compartiment dans le Amazon Simple Storage Service Guide de mise en route. Notez le nom du compartiment et la région AWS dans laquelle vous l'avez créé.
3. Sélectionnez l'onglet Autorisations, puis Stratégie de compartiment.
4. Copiez et collez la stratégie de compartiment suivante dans l'éditeur de stratégie. Remplacez *DOC-EXAMPLE-BUCKET* et *organization-id* par le nom du Amazon S3 compartiment que vous avez créé et un ID de AWS Organizations compte valide.

   (Facultatif) Remplacez *bucket-prefix* par le nom d'un Amazon S3 préfixe (sous-répertoire). Si vous n'avez pas créé de préfixe, supprimez *bucket-prefix/* depuis l'ARN dans la stratégie suivante.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "SSMBucketPermissionsCheck",
            "Effect": "Allow",
            "Principal": {
                "Service": "ssm.amazonaws.com"
            },
            "Action": "s3:GetBucketAcl",
            "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET"
        },
        {
            "Sid": " SSMBucketDelivery",
            "Effect": "Allow",
            "Principal": {
                "Service": "ssm.amazonaws.com"
            },
            "Action": "s3:PutObject",
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET/bucket-prefix/*/accountid=*/*"
            ],
            "Condition": {
                "StringEquals": {
                    "s3:x-amz-acl": "bucket-owner-full-control",
                    "s3:RequestObjectTag/OrgId": "organization-id"
                }
            }
        },
        {
            "Sid": " SSMBucketDeliveryTagging",
            "Effect": "Allow",
            "Principal": {
                "Service": "ssm.amazonaws.com"
            },
            "Action": "s3:PutObjectTagging",
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET/bucket-prefix/*/accountid=*/*"
            ]
        }
    ]
}
```

Note

La région Asie-Pacifique a été mise en ligne le 25 avril 2019. Si vous créez une synchronisation de données de ressources pour une région AWS qui est entrée en ligne depuis la région Asie-Pacifique (Hong Kong) (ap-east-1) ou une version ultérieure, vous devez saisir une entrée principale de service spécifique à la région dans la section `SSMBucketDelivery`. L'exemple suivant inclut une entrée principale de service spécifique à la région pour `ssm.ap-east-1.amazonaws.com`.

```
{
         "Sid":" SSMBucketDelivery",
         "Effect":"Allow",
         "Principal":{
            "Service":["ssm.amazonaws.com","ssm.ap-east-1.amazonaws.com"]
         },
```

Créer une synchronisation des données de ressource d'inventaire pour les comptes définis dansAWS Organizations

La procédure suivante décrit comment utiliser l' AWS CLI pour créer une synchronisation des données de ressource pour les comptes définis dans AWS Organizations . Vous devez utiliser l'AWS CLI pour effectuer cette tâche. Vous devez également effectuer cette procédure pour chaque région AWS et chaque compte définis dans AWS Organizations .

Pour créer une synchronisation des données de ressource pour les comptes définis dans AWS Organizations (AWS CLI)

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Exécutez la commande suivante pour créer une synchronisation des données de ressource pour plusieurs comptes définis dans AWS Organizations. Pour *`DOC-EXAMPLE-BUCKET`* , spécifiez le nom du Amazon S3 compartiment que vous avez créé précédemment dans cette rubrique. Si vous avez créé un préfixe (sous-répertoire) pour votre compartiment, spécifiez ces informations pour *`prefix-name`*.

```
aws ssm create-resource-data-sync --sync-name name --s3-destination "BucketName=DOC-
EXAMPLE-BUCKET,Prefix=prefix-name,SyncFormat=JsonSerDe,Region=AWS Region, for example
 us-east-2,DestinationDataSharing={DestinationDataSharingType=Organization}"
```

Configuration de la collecte d'inventaire

Cette section explique comment configurer la collecte d'inventaire sur une ou plusieurs instances gérées à l'aide de la console Systems Manager. Pour obtenir un exemple illustrant la configuration de la collecte d'inventaire à l'aide de l'AWS CLI, consultez Procédures pas à pas de l'inventaire Systems Manager (p. 837).

Lorsque vous configurez la collecte d'inventaire, vous commencez par créer une association Systems Manager State Manager. Systems Manager collecte les données d'inventaire lorsque cette association est exécutée. Si vous ne créez pas l'association en premier et essayez d'appeler le plug-in aws:softwareInventory en utilisant, par exemple, la fonctionnalité Fonctionnalité Exécuter la commande, le système renvoie l'erreur suivante :

```
The aws:softwareInventory plugin can only be invoked via ssm-associate.
```

Note

Tenez compte du comportement suivant si vous créez plusieurs associations d'inventaire pour une instance.

- Chaque instance peut se voir attribuer une association d'inventaire qui cible toutes les instances (--targets "Key=InstanceIds,Values=*").
- Chaque instance peut également se voir attribuer une association spécifique qui utilise des paires clé/valeur de balise ou un groupe de ressources AWS.
- Si une instance se voit attribuer plusieurs associations d'inventaire, le statut indique Ignored (Ignoré) pour l'association qui n'a pas été exécutée. L'association qui s'est exécutée le plus récemment affiche le statut réel de l'association d'inventaire.
- Si une instance se voit attribuer plusieurs associations d'inventaire et que chacune utilise une paire clé/valeur de balise, l'exécution de ces associations d'inventaire échoue sur l'instance en raison du conflit de balises. L'association s'exécute toujours sur les instances qui n'ont pas le conflit clé/valeur de balise.

Avant de commencer

Avant de configurer la collecte d'inventaire, effectuez les tâches suivantes.

- Mettez à jour Agent SSM sur les instances à inventorier. En exécutant la dernière version de Agent SSM, vous êtes sûr de collecter les métadonnées de tous les types d'inventaire pris en charge. Pour plus d'informations sur la mise à jour de l'Agent SSM à l'aide de State Manager, consultez Mise à jour automatique d'Agent SSM (interface de ligne de commande) (p. 1043).
- Vérifiez que vos instances respectent les prérequis de Systems Manager. Pour plus d'informations, consultez la section Prérequis d'Systems Manager (p. 15).
- (Facultatif) Créez une synchronisation de données de ressources pour stocker de manière centralisée les données d'inventaire dans un compartiment S3. La synchronisation des données de ressource met alors automatiquement à jour les données centralisées lors de la collecte de nouvelles données d'inventaire. Pour plus d'informations, consultez la section Configuration de la synchronisation de données de ressource pour l'inventaire (p. 800).
- (Facultatif) Créez un fichier JSON pour collecter l'inventaire personnalisé. Pour plus d'informations, consultez la section Utilisation de l'inventaire personnalisé (p. 826).

Effectuer l'inventaire de toutes les instances gérées dans votre compte AWS

Vous pouvez facilement inventorier toutes les instances gérées dans votre compte AWS en créant une association d'inventaire global. Une association d'inventaire global effectue les actions suivantes :

- Applique automatiquement la configuration de l'inventaire global (association) à toutes les instances gérées existant dans votre compte AWS. Les instances disposant déjà d'une association d'inventaire sont ignorées lorsque l'association d'inventaire global est appliquée et s'exécute. Lorsqu'une instance est ignorée, le message d'état détaillé indique `Overridden By Explicit Inventory Association` . Ces instances sont ignorées par l'association globale, mais elles continuent de générer un inventaire lorsqu'elles exécutent l'association d'inventaire qui leur est attribuée.
- Ajoute automatiquement les instances récemment créées dans votre compte AWS à l'association d'inventaire global.

Note

- Si une instance est configurée pour l'association d'inventaire global et que vous lui attribuez une association spécifique, l'inventaire Systems Manager annule la priorité de l'association globale et applique l'association spécifique.
- Les associations d'inventaire global sont disponibles dans Agent SSM version 2.0.790.0 ou ultérieure. Pour plus d'informations sur la mise à jour de l'Agent SSM sur vos instances, consultez Mettre à jour l'Agent SSM à l'aide de Fonctionnalité Exécuter la commande (p. 948).

Configuration de la collecte d'inventaire en un clic (console)

Appliquez la procédure suivante pour configurer l'inventaire Systems Manager pour toutes les instances gérées dans votre compte AWS et dans une seule région AWS.

Pour configurer l'ensemble de vos instances gérées dans la région actuelle pour l'inventaire Systems Manager

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Inventaire.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Inventaire dans le panneau de navigation.
3. Dans la carte Managed instances with inventory enabled (Instances gérées avec l'inventaire activé), choisissez Click here to enable inventory on all instances (Cliquez ici pour activer l'inventaire sur toutes les instances).

   En cas de réussite, la console affiche le message suivant.

   En fonction du nombre d'instances gérées dans votre compte, l'application de l'association d'inventaire global peut prendre plusieurs minutes. Patientez quelques minutes, puis actualisez la page. Vérifiez que le graphique change pour montrer que l'inventaire est configuré sur l'ensemble de vos instances gérées.

Configuration de la collecte à l'aide de la console

Cette section inclut des informations sur la configuration de l'inventaire Systems Manager pour qu'il collecte les métadonnées de vos instances gérées à l'aide de la console Systems Manager. Vous pouvez collecter

rapidement les métadonnées de toutes les instances d'un compte AWS spécifique (et de toutes les futures instances susceptibles d'être créées dans ce compte) ou vous pouvez collecter de manière sélective les données d'inventaire à l'aide de balises ou d'une instance IDs .

Pour configurer la collecte d'inventaire

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Inventaire.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Inventaire dans le panneau de navigation.
3. Choisissez Setup Inventory (Configurer l'inventaire).
4. Dans la section Targets (Cibles), identifiez les instances sur lesquelles vous souhaitez exécuter cette opération en choisissant l'un des éléments suivants.

   - Selecting all managed instances in this account (Sélection de toutes les instances gérées dans ce compte) - Cette option sélectionne toutes les instances gérées ne disposant d'aucune association d'inventaire existante. Si vous choisissez cette option, les instances disposant déjà d'associations d'inventaire sont ignorées lors de la collecte d'inventaire, et s'affichent avec un statut Skipped (Ignoré) dans les résultats d'inventaire. Pour plus d'informations, consultez la section Effectuer l'inventaire de toutes les instances gérées dans votre compte AWS (p. 807).
   - Specifying a tag (Spécification d'une balise) - Cette option vous permet de spécifier une balise unique pour identifier les instances de votre compte à partir desquelles vous souhaitez collecter l'inventaire. Si vous utilisez une balise, toutes les instances créées à l'avenir avec la même balise généreront également des rapports d'inventaires. S'il existe déjà une association d'inventaire avec toutes les instances, l'utilisation d'une balise pour sélectionner des instances spécifiques en tant que cible pour un autre inventaire remplace l'appartenance de l'instance dans le groupe cible All managed instances (Toutes les instances gérées). Les instances avec la balise spécifiée sont ignorées lors des futures collectes d'inventaire à partir de All managed instances (Toutes les instances gérées).
   - Manually selecting instances (Sélection manuelle des instances) - Cette option vous permet de choisir des instances gérées spécifiques dans votre compte. Choisir explicitement des instances spécifiques à l'aide de cette option remplace les associations d'inventaire sur la cible All managed instances (Toutes les instances gérées). L'instance est ignorée lors des futures collectes d'inventaire à partir de All managed instances (Toutes les instances gérées).

     Note

     Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.
5. Dans la section Schedule (Programme), choisissez la fréquence à laquelle le système doit collecter les métadonnées d'inventaire à partir de vos instances.
6. Dans la section Parameters (Paramètres), utilisez les listes pour activer ou désactiver les différents types de collecte d'inventaire. Consultez les exemples suivants pour créer une recherche d'inventaire pour des fichiers ou le registre Windows.

   #### Files

   - Sous Linux, collectez les métadonnées des fichiers .sh dans le répertoire `/home/ec2-user`, en excluant tous les sous-répertoires.

     ```
     [{"Path":"/home/ec2-user","Pattern":["*.sh", "*.sh"],"Recursive":false}]
     ```

- Sous Windows, collectez les métadonnées de tous les fichiers .exe du dossier Program Files, en incluant les sous-répertoires de manière récursive.

```
[{"Path":"C:\Program Files","Pattern":["*.exe"],"Recursive":true}]
```

- Sous Windows, collectez les métadonnées de modèles de journaux spécifiques.

```
[{"Path":"C:\ProgramData\Amazon","Pattern":["*amazon*.log"],"Recursive":true}]
```

- Limitez le nombre de répertoires lors de l'exécution d'une collection récursive.

```
[{"Path":"C:\Users","Pattern":["*.ps1"],"Recursive":true, "DirScanLimit": 1000}]
```

Registre Windows

- Collectez toutes les clés et valeurs de manière récursive pour un chemin spécifique.

```
[{"Path":"HKEY_LOCAL_MACHINE\SOFTWARE\Amazon","Recursive": true}]
```

- Collectez toutes les clés et valeurs pour un chemin spécifique (recherche récursive désactivée).

```
[{"Path":"HKEY_LOCAL_MACHINE\SOFTWARE\Intel\PSIS\PSIS_DECODER", "Recursive": false}]
```

- Collectez une clé spécifique à l'aide de l'option `ValueNames`.

```
{"Path":"HKEY_LOCAL_MACHINE\SOFTWARE\Amazon\MachineImage","ValueNames":["AMIName"]}
```

Pour plus d'informations sur la collecte de l'inventaire de fichiers et du registre Windows, consultez Utilisation de l'inventaire de fichiers et du registre Windows (p. 798).

7. Dans la section Avancé, choisissez Sync inventory execution logs to an S3 bucket (Synchroniser les journaux d'exécution d'inventaire de synchronisation sur un compartiment S3) si vous souhaitez stocker le statut d'exécution d'association dans un compartiment S3.
8. Choisissez Setup Inventory (Configurer l'inventaire). Systems Manager crée une association State Manager et exécute immédiatement l'inventaire sur les instances.
9. Dans le panneau de navigation, choisissez State Manager. Vérifiez qu'une nouvelle association utilisant le document AWS-GatherSoftwareInventory a été créée. La planification de l'association utilise une expression rate. Vérifiez également que le champ Status (Statut) affiche la valeur Success (Réussite). Si vous avez choisi l'option Sync inventory execution logs to an S3 bucket (Synchroniser les journaux d'exécution d'inventaire sur un compartiment S3), vous pouvez afficher les données des journaux dans Amazon S3 après quelques minutes. Si vous souhaitez afficher les données d'inventaire pour une instance spécifique, choisissez Managed Instances (Instances gérées) dans le panneau de navigation.
10. Choisissez une instance, puis choisissez View details (Afficher les détails).
11. Sur la page des détails de l'instance, choisissez Inventory (Inventaire). Utilisez les listes Inventory type (Type d'inventaire) pour filtrer l'inventaire.

Utilisation des données d'inventaire Systems Manager

Cette section inclut des rubriques qui décrivent comment interroger et regrouper les données d'inventaire.

Rubriques

- Interrogation des données d'inventaire à partir de plusieurs régions et comptes (p. 811)
- Interrogation d'une collecte d'inventaire à l'aide de filtres (p. 816)
- Agrégation des données d'inventaire (p. 816)

Interrogation des données d'inventaire à partir de plusieurs régions et comptes

L'inventaire AWS Systems Manager s'intègre à Amazon Athena pour vous aider à interroger les données d'inventaire provenant de plusieurs régions et comptes AWS. L'intégration d'Athena utilise la synchronisation de données de ressources pour que vous puissiez consulter les données d'inventaire provenant de toutes vos instances gérées sur la page Inventory Detail View (Vue des détails d'inventaire), dans la console AWS Systems Manager.

> **Important**
>
> Cette fonction utilise AWS Glue pour analyser les données de votre compartiment Amazon Simple Storage Service (Amazon S3), et Amazon Athena pour interroger les données. En fonction de la quantité de données analysées et interrogées, l'utilisation de ces services peut vous être facturée. Avec AWS Glue, vous payez un taux horaire facturé à la seconde pour les analyseurs (détection des données) et les tâches ETL (traitement et chargement des données). Avec Athena, vous êtes facturé en fonction de la quantité de données analysées par chaque requête. Nous vous incitons à consulter les consignes de tarification de ces services avant d'utiliser l'intégration d'Amazon Athena avec l'inventaire Systems Manager. Pour plus d'informations, consultez Tarification Amazon Athena et Tarification AWS Glue.

Vous pouvez afficher les données d'inventaire sur la page Inventory Detail View (Vue des détails d'inventaire) dans toutes les régions AWS où Amazon Athena est disponible. Pour obtenir une liste des régions prises en charge, consultez Points de terminaison de service Amazon Athena dans le Référence générale d'Amazon Web Services.

Avant de commencer

L'intégration d'Athena utilise la synchronisation de données de ressources. Vous devez installer et configurer la synchronisation de données de ressources pour utiliser cette fonction. Pour plus d'informations, consultez la section Configuration de la synchronisation de données de ressource pour l'inventaire (p. 800).

De plus, sachez que la page Inventory Detail View (Vue des détails d'inventaire) affiche les données d'inventaire pour le propriétaire du compartiment S3 central utilisé par la synchronisation de données de ressources. Si vous n'êtes pas le propriétaire du compartiment S3 central, vous ne verrez pas les données d'inventaire sur la page Inventory Detail View (Vue des détails d'inventaire).

Configuration de l'accès

Avant de pouvoir interroger et afficher des données de plusieurs comptes et régions sur la page Inventory Detail View (Vue des détails d'inventaire) dans la console Systems Manager, vous devez configurer votre compte utilisateur AWS Identity and Access Management (IAM). La procédure suivante explique comment utiliser la console IAM pour configurer votre compte utilisateur IAM afin de pouvoir afficher les données d'inventaire sur la page Inventory Detail View (Vue des détails d'inventaire).

Pour configurer l'accès à la page de la vue détaillée de l'inventaire

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation, choisissez Utilisateurs, puis choisissez le compte utilisateur que vous voulez configurer. La page Récapitulatif s'ouvre.
3. Sous l'onglet Autorisations, choisissez Ajouter des autorisations.

4. Sur la page Grant permissions (Accorder des autorisations), choisissez Attacher directement les stratégies existantes.
5. Dans le champ de recherche, recherchez AWSQuicksightAthenaAccess .
6. Choisissez l'option en regard de cette stratégie, puis choisissez Next (Suivant) : Passez en revue.
7. Choisissez Add permissions (Ajouter des autorisations).
8. Choisissez de nouveau le nom d'utilisateur pour revenir à la page Récapitulatif.
9. Maintenant, ajoutez une stratégie en ligne afin de permettre à AWS Glue d'analyser vos données d'inventaire. Sous l'onglet Autorisations, à droite de la page, choisissez Add inline policy (Ajouter une stratégie en ligne). La page Créer une stratégie s'ouvre.
10. Choisissez l'onglet JSON.
11. Supprimez le texte JSON existant dans l'éditeur, puis copiez et collez la stratégie suivante dans l'éditeur JSON.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "glue:GetCrawlers",
                "glue:GetCrawler",
                "glue:GetTables",
                "glue:StartCrawler",
                "glue:CreateCrawler"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:PassRole",
                "iam:CreateRole",
                "iam:AttachRolePolicy"
            ],
            "Resource": [
                "arn:aws:iam::account_ID:role/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:CreatePolicy"
            ],
            "Resource": [
                "arn:aws:iam::account_ID:policy/*"
            ]
        }
    ]
}
```

Note

(Facultatif) Si le compartiment Amazon Simple Storage Service (Amazon S3) utilisé pour stocker les données d'inventaire est chiffré à l'aide d'AWS Key Management Service, vous devez également ajouter le bloc suivant à la stratégie.

```
{
    "Effect": "Allow",
    "Action": [
```

```
            "kms:Decrypt"
        ],
        "Resource": [
            "arn:aws:kms:AWS_Region:account_ID:key/key_ARN"
        ]
}
```

Si vous collez ce bloc après le dernier bloc de la stratégie, veillez à séparer les blocs par une virgule (,).

12. Sur la page Examiner une stratégie, entrez un nom dans le champ Nom.
13. Choisissez Create policy (Créer une stratégie).

Important

Lorsque vous choisissez une synchronisation de données de ressource sur la page Inventory Detail View (Vue des détails d'inventaire), Systems Manager crée automatiquement le rôle Amazon-GlueServiceRoleForSSM. Ce rôle permet à AWS Glue d'accéder au compartiment S3 pour la synchronisation de données de ressources. Systems Manager attache automatiquement les stratégies suivantes à ce rôle :

- Nom du Amazon S3 compartiment Amazon-GlueServicePolicyForSSM-{} : Cette stratégie permet la communication entre AWS Glue et Systems Manager l'inventaire .
- AWSGlueServiceRole: Il s'agit d'une stratégie gérée par AWS qui permet d'accéder à AWS Glue .

S'il existe déjà une stratégie nommée Amazon-GlueServicePolicyForSSM-{nom du compartiment S3} dans votre compte utilisateur IAM et que cette stratégie n'est pas attachée au rôle Amazon-GlueServiceRoleForSSM, le système renvoie une erreur. Pour résoudre ce problème, utilisez la console IAM pour vérifier que le contenu de la stratégie Amazon-GlueServicePolicyForSSM-{nom du compartiment Amazon S3} correspond bien à la stratégie en ligne de cette procédure. Ensuite, attachez la stratégie au rôle Amazon-GlueServiceRoleForSSM.

Interrogation des données sur la page Vue détaillée d'inventaire

Utilisez la procédure suivante pour afficher des données d'inventaire provenant de plusieurs régions et comptes AWS sur la page Detailed View (Vue détaillée) de l’inventaire Systems Manager.

Important

La page Detailed View (Vue détaillée) de l’inventaire est uniquement disponible dans les régions AWS qui proposent Amazon Athena. Si les onglets suivants ne sont pas affichés sur la page d'inventaire Systems Manager, cela signifie qu'Athena n'est pas disponible dans la région et que vous ne pouvez pas utiliser la Detailed View (Vue détaillée) pour interroger les données.

Pour afficher les données d'inventaire à partir de plusieurs régions et comptes dans la console AWS Systems Manager

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Inventaire.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Inventaire dans le panneau de navigation.
3. Choisissez l'onglet Detailed View (Vue détaillée).

4. Choisissez la synchronisation de données de ressources pour laquelle vous souhaitez rechercher des données.

5. Dans la liste Type d'inventaire, choisissez le type de données d'inventaire que vous souhaitez interroger, puis appuyez sur Enter.

6. Pour filtrer les données, choisissez la barre de filtre, puis choisissez une option de filtre.

L'exemple suivant montre les données AWSComponent d'inventaire filtrées sur la région us-east-2.

Vous pouvez utiliser le bouton Export to CSV (Exporter au format CSV) pour afficher l'ensemble de requêtes actuel dans un tableur tel que Microsoft Excel. Vous pouvez également utiliser les boutons Query History (Historique de requête) et Run Advanced Queries (Exécuter des requêtes avancées) pour afficher les détails d'historique et interagir avec vos données dans Amazon Athena.

Modification de la planification de l'analyseur AWS Glue

AWS Glue analyse les données d'inventaire dans le compartiment S3 central deux fois par jour, par défaut. Si vous modifiez fréquemment les types de données à collecter sur vos instances, vous préférerez peut-être analyser les données plus fréquemment, comme cela est décrit dans la procédure suivante.

> **Important**
>
> AWS Glue impute des frais à votre compte sur la base d'un taux horaire facturé à la seconde pour les analyseurs (détection des données) et les tâches ETL (traitement et chargement des données). Avant de modifier la planification de l'analyseur, consultez la page Tarification AWS Glue.

Pour modifier la planification de l'analyseur de données d'inventaire

1. Ouvrez la console AWS Glue à l'adresse https://console.aws.amazon.com/glue/.
2. Dans le panneau de navigation, choisissez Crawlers (Analyseurs).
3. Dans la liste des analyseurs, choisissez l'option à côté de l'analyseur de données d'inventaire Systems Manager. Le nom de l'analyseur utilise le format suivant :

   AWSSystemsManager-*DOC-EXAMPLE-BUCKET*-*Region*-*AWS_account_ID*
4. Choisissez Action, puis Modifier un analyseur.
5. Dans le panneau de navigation, choisissez Planification.

6. Dans le champ Expression cron, spécifiez une nouvelle planification à l'aide d'un format cron. Pour plus d'informations sur le format cron, consultez Planifications temporelles pour les tâches et les analyseurs dans le Manuel du développeur AWS Glue.

   Important

   Vous pouvez suspendre l'analyseur pour arrêter les frais imputés par AWS Glue. Si vous suspendez l'analyseur ou si vous modifiez la fréquence pour analyser moins souvent les données, la page Detailed View (Vue détaillée) d'inventaire peut afficher des données qui ne sont pas à jour.

Interrogation d'une collecte d'inventaire à l'aide de filtres

Une fois que vous avez collecté les données d'inventaire, vous pouvez utiliser les fonctionnalités de filtre dans Systems Manager pour interroger une liste d'instances gérées respectant certains critères de filtre.

Pour interroger des instances d'après des filtres d'inventaire

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Inventaire.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Inventaire dans le panneau de navigation.
3. Dans la section Filter by resource groups, tags or inventory types (Filtrer par groupes de ressources, balises ou types d'inventaire), choisissez la zone de filtre. Une liste de filtres prédéfinis s'affiche.
4. Choisissez un attribut sur lequel filtrer. Par exemple, choisissez AWS:Application. Si vous y êtes invité, choisissez un attribut secondaire sur lequel filtrer. Par exemple, choisissez AWS:Application.Name.
5. Choisissez un délimiteur dans la liste. Par exemple, choisissez Begin with. Une zone de texte s'affiche dans le filtre.
6. Saisissez une valeur dans la zone de texte. Par exemple, saisissez Amazon (l'Agent SSM est nommé Agent SSM Amazon).
7. Appuyez sur Enter. Le système renvoie une liste des instances gérées qui incluent un nom d'application commençant par le mot Amazon.

   Note

   Vous pouvez combiner plusieurs filtres pour affiner votre recherche.

Agrégation des données d'inventaire

Une fois que vous avez configuré vos instances gérées pour l'inventaire AWS Systems Manager, vous pouvez afficher les décomptes agrégés des données d'inventaire. Par exemple, supposons que vous avez configuré des dizaines ou des centaines d'instances gérées pour collecter le type d'inventaire AWS:Application. En utilisant les informations de cette section, vous pouvez voir le décompte exact des instances configurées pour collecter ces données.

Vous pouvez également voir des détails d'inventaire spécifiques en effectuant une agrégation sur un type de données. Par exemple, le type d'inventaire AWS:InstanceInformation collecte les informations de plateforme de système d'exploitation avec le type de données `Platform`. En agrégeant les données sur le type de données `Platform`, vous pouvez rapidement voir le nombre d'instances qui exécutent Windows et le nombre d'instances qui exécutent Linux.

Les procédures de cette section expliquent comment afficher les décomptes agrégés de données d'inventaire à l'aide de l'AWS CLI. Vous pouvez également consulter les décomptes agrégés préconfigurés

dans la console AWS Systems Manager sur la page Inventaire. Ces tableaux de bord préconfigurés sont appelés analyses d'inventaire et ils permettent de remédier en un clic à vos problèmes de configuration d'inventaire.

Notez les détails importants suivants relatifs aux décomptes d'agrégation des données d'inventaire :

- L'inventaire Systems Manager stocke les données d'inventaire pendant 30 jours. Cela signifie que les décomptes agrégés d'inventaire incluent toutes les données collectées au cours des 30 derniers jours.
- L'inventaire montre les données qui ont été envoyées par une instance au cours de son cycle de vie. Si une instance a été préalablement configurée pour signaler un type de données d'inventaire spécifique, par exemple AWS:Network, et que vous modifiez ultérieurement cette configuration pour arrêter la collecte de ce type, les décomptes d'agrégation continuent de montrer les données AWS:Network jusqu'à ce que l'instance soit mise hors service.
- Si une instance a été préalablement configurée pour collecter des données d'inventaire et que vous mettez hors service cette instance, les décomptes d'inventaire continuent de montrer pendant 30 jours les données pour l'instance supprimée.

Pour obtenir des informations sur la façon de configurer et de collecter rapidement des données d'inventaire à partir de toutes les instances dans un compte AWS spécifique (et de toutes les instances futures susceptibles d'être créées dans ce compte), consultez Configuration de la collecte à l'aide de la console (p. 808).

Rubriques

- Agrégation des données d'inventaire pour afficher les décomptes d'instances qui collectent des types spécifiques de données (p. 817)
- Agrégation des données d'inventaire avec des groupes pour voir quelles instances sont ou non configurées pour collecter un type d'inventaire (p. 821)

Agrégation des données d'inventaire pour afficher les décomptes d'instances qui collectent des types spécifiques de données

Vous pouvez utiliser l'action GetInventory d'API pour afficher les décomptes agrégés des instances qui collectent un ou plusieurs types d'inventaire et types de données. Par exemple, le type d'inventaire AWS:InstanceInformation vous permet d'afficher un agrégat de systèmes d'exploitation à l'aide de l'action dGetInventory'API avec le type de données AWS:InstanceInformation.PlatformType. Voici un exemple de commande AWS CLI et de sortie :

```
aws ssm get-inventory --aggregators "Expression=AWS:InstanceInformation.PlatformType"
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "Entities":[
      {
         "Data":{
            "AWS:InstanceInformation":{
               "Content":[
                  {
                     "Count":"7",
                     "PlatformType":"windows"
                  },
                  {
                     "Count":"5",
                     "PlatformType":"linux"
                  }
               ]
```

```
            }
        }
    }
  ]
}
```

Mise en route

Déterminez les types d'inventaire et les types de données pour lesquels vous souhaitez afficher des décomptes. Vous pouvez afficher la liste des types d'inventaire et des types de données qui prennent en charge l'agrégation en exécutant la commande suivante dans l'AWS CLI :

```
aws ssm get-inventory-schema --aggregator
```

Cette commande renvoie la liste JSON des types d'inventaire et des types de données qui prennent en charge l'agrégation. Le TypeName champ affiche les types d'inventaire pris en charge. Et le champ Name montre chaque type de données. Par exemple, dans la liste suivante, le type d'inventaire AWS:Application inclut des types de données pour Name et Version.

```
{
    "Schemas": [
        {
            "TypeName": "AWS:Application",
            "Version": "1.1",
            "DisplayName": "Application",
            "Attributes": [
                {
                    "DataType": "STRING",
                    "Name": "Name"
                },
                {
                    "DataType": "STRING",
                    "Name": "Version"
                }
            ]
        },
        {
            "TypeName": "AWS:InstanceInformation",
            "Version": "1.0",
            "DisplayName": "Platform",
            "Attributes": [
                {
                    "DataType": "STRING",
                    "Name": "PlatformName"
                },
                {
                    "DataType": "STRING",
                    "Name": "PlatformType"
                },
                {
                    "DataType": "STRING",
                    "Name": "PlatformVersion"
                }
            ]
        },
        {
            "TypeName": "AWS:ResourceGroup",
            "Version": "1.0",
            "DisplayName": "ResourceGroup",
            "Attributes": [
                {
                    "DataType": "STRING",
                    "Name": "Name"
```

```
                }
            ]
        },
        {
            "TypeName": "AWS:Service",
            "Version": "1.0",
            "DisplayName": "Service",
            "Attributes": [
                {
                    "DataType": "STRING",
                    "Name": "Name"
                },
                {
                    "DataType": "STRING",
                    "Name": "DisplayName"
                },
                {
                    "DataType": "STRING",
                    "Name": "ServiceType"
                },
                {
                    "DataType": "STRING",
                    "Name": "Status"
                },
                {
                    "DataType": "STRING",
                    "Name": "StartType"
                }
            ]
        },
        {
            "TypeName": "AWS:WindowsRole",
            "Version": "1.0",
            "DisplayName": "WindowsRole",
            "Attributes": [
                {
                    "DataType": "STRING",
                    "Name": "Name"
                },
                {
                    "DataType": "STRING",
                    "Name": "DisplayName"
                },
                {
                    "DataType": "STRING",
                    "Name": "FeatureType"
                },
                {
                    "DataType": "STRING",
                    "Name": "Installed"
                }
            ]
        }
    ]
}
```

Vous pouvez agréger les données pour l'un quelconque des types d'inventaire répertoriés en créant une commande qui utilise la syntaxe suivante :

```
aws ssm get-inventory --aggregators "Expression=InventoryType.DataType"
```

Voici quelques exemples d'.

Exemple 1

Cet exemple agrège un décompte des rôles Windows utilisés par vos instances.

```
aws ssm get-inventory --aggregators "Expression=AWS:WindowsRole.Name"
```

Exemple 2

Cet exemple agrège un décompte des applications installées sur vos instances.

```
aws ssm get-inventory --aggregators "Expression=AWS:Application.Name"
```

Combinaison de plusieurs agrégateurs

Vous pouvez également combiner plusieurs types d'inventaire et types de données en une seule commande pour mieux comprendre les données. Voici quelques exemples d'.

Exemple 1

Cet exemple agrège un décompte des types de système d'exploitation utilisés par vos instances. Il renvoie également le nom spécifique de ces systèmes d'exploitation.

```
aws ssm get-inventory --aggregators '[{"Expression":
 "AWS:InstanceInformation.PlatformType", "Aggregators":[{"Expression":
 "AWS:InstanceInformation.PlatformName"}]}]'
```

Exemple 2

Cet exemple agrège un décompte des applications qui s'exécutent sur vos instances et de la version spécifique de chaque application.

```
aws ssm get-inventory --aggregators '[{"Expression": "AWS:Application.Name", "Aggregators":
[{"Expression": "AWS:Application.Version"}]}]'
```

Si vous préférez, vous pouvez créer une expression d'agrégation avec un ou plusieurs types d'inventaire et types de données dans un fichier JSON et appeler ce fichier à partir de l'AWS CLI. Le code JSON figurant dans le fichier doit utiliser la syntaxe suivante :

```
[
       {
           "Expression": "string",
           "Aggregators": [
               {
                  "Expression": "string"
               }
           ]
       }
]
```

Vous devez enregistrer le fichier avec l'extension de fichier .json.

Voici un exemple qui utilise plusieurs types d'inventaire et types de données.

```
[
       {
           "Expression": "AWS:Application.Name",
           "Aggregators": [
               {
                   "Expression": "AWS:Application.Version",
                   "Aggregators": [
```

```
                    {
                    "Expression": "AWS:InstanceInformation.PlatformType"
                    }
                  ]
                }
            ]
        }
]
```

Utilisez la commande suivante pour appeler le fichier à partir de l'AWS CLI.

```
aws ssm get-inventory --aggregators file://file_name.json
```

La commande renvoie des informations semblables aux suivantes :

```
{"Entities":
 [
   {"Data":
     {"AWS:Application":
       {"Content":
         [
           {"Count": "3",
            "PlatformType": "linux",
            "Version": "2.6.5",
            "Name": "audit-libs"},
           {"Count": "2",
            "PlatformType": "windows",
            "Version": "2.6.5",
            "Name": "audit-libs"},
           {"Count": "4",
            "PlatformType": "windows",
            "Version": "6.2.8",
            "Name": "microsoft office"},
           {"Count": "2",
            "PlatformType": "windows",
            "Version": "2.6.5",
            "Name": "chrome"},
           {"Count": "1",
            "PlatformType": "linux",
            "Version": "2.6.5",
            "Name": "chrome"},
           {"Count": "2",
            "PlatformType": "linux",
            "Version": "6.3",
            "Name": "authconfig"}
         ]
       }
     },
    "ResourceType": "ManagedInstance"}
 ]
}
```

Agrégation des données d'inventaire avec des groupes pour voir quelles instances sont ou non configurées pour collecter un type d'inventaire

Les groupes vous permettent de voir rapidement un décompte des instances gérées qui sont ou ne sont pas configurées pour collecter un ou plusieurs types d'inventaire. Avec les groupes, vous spécifiez un ou plusieurs types d'inventaire et un filtre qui utilise l'opérateur `exists`.

Par exemple, supposons que vous avez quatre instances gérées configurées pour collecter les types d'inventaire suivants :

- Instance 1 : Application:AWS
- Instance 2 : Fichier:AWS
- Instance 3 : </s><s id="20-1-1">Instance 4 :
- Instance 4: Réseau:AWS

Vous pouvez exécuter la commande suivante à partir de l'AWS CLI pour voir combien d'instances sont configurées pour collecter les types d'inventaire AWS:Application et AWS:File. La réponse renvoie également un décompte des instances qui ne sont pas configurées pour collecter ces deux types d'inventaire.

```
aws ssm get-inventory --aggregators
 'Groups=[{Name=ApplicationAndFile,Filters=[{Key=TypeName,Values=[AWS:Application],Type=Exists},
{Key=TypeName,Values=[AWS:File],Type=Exists}]}]'
```

La réponse la commande montre qu'une seule instance gérée est configurée pour collecter les deux types d'inventaire AWS:Application et AWS:File.

```
{
   "Entities":[
      {
         "Data":{
            "ApplicationAndFile":{
               "Content":[
                  {
                     "notMatchingCount":"3"
                  },
                  {
                     "matchingCount":"1"
                  }
               ]
            }
         }
      }
   ]
}
```

> **Note**
>
> Les groupes ne renvoient pas de décomptes de type de données. En outre, vous ne pouvez pas explorer en détail les résultats pour voir les IDs instances qui sont ou non configurées pour collecter le type d'inventaire.

Si vous préférez, vous pouvez créer une expression d'agrégation avec un ou plusieurs types d'inventaire dans un fichier JSON et appeler ce fichier à partir de l'AWS CLI. Le code JSON figurant dans le fichier doit utiliser la syntaxe suivante :

```
{
   "Aggregators":[
      {
         "Groups":[
            {
               "Name":"Name",
               "Filters":[
                  {
                     "Key":"TypeName",
                     "Values":[
                        "Inventory_type"
                     ],
                     "Type":"Exists"
```

```
                    },
                    {
                        "Key":"TypeName",
                        "Values":[
                            "Inventory_type"
                        ],
                        "Type":"Exists"
                    }
                ]
            }
        ]
    }
    ]
}
```

Vous devez enregistrer le fichier avec l'extension de fichier .json.

Utilisez la commande suivante pour appeler le fichier à partir de l'AWS CLI.

```
aws ssm get-inventory --cli-input-json file://file_name.json
```

Exemples supplémentaires

Les exemples suivants vous montrent comment agréger les données d'inventaire pour voir quelles instances gérées sont ou non configurées pour collecter les types d'inventaire spécifiés. Ces exemples utilisent l'AWS CLI. Chaque exemple inclut une commande complète avec des filtres que vous pouvez exécuter à partir de la ligne de commande et un exemple de fichier input.json si vous préférez entrer les informations dans un fichier.

Exemple 1

Cet exemple regroupe un décompte des instances qui sont ou ne sont pas configurées pour collecter le type d'inventaire AWS:Application ou AWS:File.

Exécutez la commande suivante à partir de l'AWS CLI.

```
aws ssm get-inventory --aggregators
 'Groups=[{Name=ApplicationORFile,Filters=[{Key=TypeName,Values=[AWS:Application,
 AWS:File],Type=Exists}]}]'
```

Si vous préférez utiliser un fichier, copiez et collez l'exemple suivant dans un fichier et enregistrez-le sous le nom input.json.

```
{
    "Aggregators":[
        {
            "Groups":[
                {
                    "Name":"ApplicationORFile",
                    "Filters":[
                        {
                            "Key":"TypeName",
                            "Values":[
                                "AWS:Application",
                                "AWS:File"
                            ],
                            "Type":"Exists"
                        }
                    ]
                }
            ]
```

```
      }
   ]
}
```

Exécutez la commande suivante à partir de l'AWS CLI.

```
aws ssm get-inventory --cli-input-json file://input.json
```

La commande renvoie des informations semblables aux suivantes :

```
{
   "Entities":[
      {
         "Data":{
            "ApplicationORFile":{
               "Content":[
                  {
                     "notMatchingCount":"1"
                  },
                  {
                     "matchingCount":"3"
                  }
               ]
            }
         }
      }
   ]
}
```

Exemple 2

Cet exemple regroupe un décompte des instances qui sont ou ne sont pas configurées pour collecter les types d'inventaire AWS:Application, AWS:File et AWS:Network.

Exécutez la commande suivante à partir de l'AWS CLI.

```
aws ssm get-inventory --aggregators
 'Groups=[{Name=Application,Filters=[{Key=TypeName,Values=[AWS:Application],Type=Exists}]},
 {Name=File,Filters=[{Key=TypeName,Values=[AWS:File],Type=Exists}]},
 {Name=Network,Filters=[{Key=TypeName,Values=[AWS:Network],Type=Exists}]}]'
```

Si vous préférez utiliser un fichier, copiez et collez l'exemple suivant dans un fichier et enregistrez-le sous le nom input.json.

```
{
   "Aggregators":[
      {
         "Groups":[
            {
               "Name":"Application",
               "Filters":[
                  {
                     "Key":"TypeName",
                     "Values":[
                        "AWS:Application"
                     ],
                     "Type":"Exists"
                  }
               ]
            },
```

```
            {
               "Name":"File",
               "Filters":[
                  {
                     "Key":"TypeName",
                     "Values":[
                        "AWS:File"
                     ],
                     "Type":"Exists"
                  }
               ]
            },
            {
               "Name":"Network",
               "Filters":[
                  {
                     "Key":"TypeName",
                     "Values":[
                        "AWS:Network"
                     ],
                     "Type":"Exists"
                  }
               ]
            }
         ]
      }
   ]
}
```

Exécutez la commande suivante à partir de l'AWS CLI.

```
aws ssm get-inventory --cli-input-json file://input.json
```

La commande renvoie des informations semblables aux suivantes :

```
{
   "Entities":[
      {
         "Data":{
            "Application":{
               "Content":[
                  {
                     "notMatchingCount":"2"
                  },
                  {
                     "matchingCount":"2"
                  }
               ]
            },
            "File":{
               "Content":[
                  {
                     "notMatchingCount":"2"
                  },
                  {
                     "matchingCount":"2"
                  }
               ]
            },
            "Network":{
               "Content":[
                  {
                     "notMatchingCount":"3"
```

```
                    },
                    {
                       "matchingCount":"1"
                    }
                ]
            }
        }
    }
  ]
}
```

Utilisation de l'inventaire personnalisé

Vous pouvez attribuer toutes les métadonnées souhaitées à vos instances en créant un inventaire personnalité. Par exemple, supposons que vous gériez un grand nombre de serveurs dans des racks dans votre centre de données et que ces serveurs ont été configurés en tant qu'instances gérées Systems Manager. Vous stockez actuellement des informations sur l'emplacement des racks de serveurs dans une feuille de calcul. Avec un inventaire personnalisé, vous pouvez spécifier l'emplacement des racks de chaque instance en tant que métadonnées de l'instance. Lors de la collecte d'inventaire à l'aide de Systems Manager, ces métadonnées sont collectées avec les autres métadonnées d'inventaire. Vous pouvez ensuite transférer toutes les métadonnées d'inventaire vers un compartiment Amazon S3 central à l'aide de la synchronisation de données de ressources et interroger les données.

> **Note**
>
> Systems Manager prend en charge 20 types d'inventaires personnalisés au maximum par compte AWS.

Pour attribuer un inventaire personnalisé à une instance, vous pouvez utiliser l'action Systems Manager PutInventory d'API , comme décrit dans Procédure : Attribuer des métadonnées d'inventaire personnalisé à une instance (p. 838) . Sinon, vous pouvez créer un fichier JSON d'inventaire personnalisé et le charger sur l'instance. Cette section explique comment créer le fichier JSON.

L'exemple suivant de fichier JSON avec inventaire personnalisé spécifie les informations sur les racks concernant un serveur sur site. Cet exemple spécifie un type de données d'inventaire personnalisé (`"TypeName": "Custom:RackInformation"`), avec plusieurs entrées sous `Content` qui décrivent les données.

```
{
    "SchemaVersion": "1.0",
    "TypeName": "Custom:RackInformation",
    "Content": {
        "Location": "US-EAST-02.CMH.RACK1",
        "InstalledTime": "2016-01-01T01:01:01Z",
        "vendor": "DELL",
        "Zone" : "BJS12",
        "TimeZone": "UTC-8"
      }
 }
```

Vous pouvez également spécifier des entrées distinctes dans la section `Content`, comme illustré dans l'exemple suivant.

```
{
    "SchemaVersion": "1.0",
"TypeName": "Custom:PuppetModuleInfo",
    "Content": [{
        "Name": "puppetlabs/aws",
        "Version": "1.0"
      },
```

```
      {
        "Name": "puppetlabs/dsc",
        "Version": "2.0"
      }
    ]
}
```

Le schéma JSON pour l'inventaire personnalisé nécessite les sections SchemaVersion , et Content, mais vous pouvez définir les informations de ces sections.TypeName

```
{
    "SchemaVersion": "user_defined",
    "TypeName": "Custom:user_defined",
    "Content": {
        "user_defined_attribute1": "user_defined_value1",
        "user_defined_attribute2": "user_defined_value2",
        "user_defined_attribute3": "user_defined_value3",
        "user_defined_attribute4": "user_defined_value4"
      }
 }
```

TypeNameest limité à 100 caractères. La TypeName section doit également commencer par Custom. Par exemple, Custom:PuppetModuleInfo. et personnalisé `Data` vous spécifiez doit commencer par une majuscule. Les exemples suivants entraîneraient une exception : « CUSTOM:RackInformation », « custom:rackinformation ».

La section Content inclut les attributs et `data`. Ces éléments ne sont pas sensibles à la casse. Par contre, si vous définissez un attribut (par exemple: « Fournisseur » : « DELL »), vous devez faire référence à cet attribut de manière cohérente dans vos fichiers d'inventaire personnalisés. Si vous spécifiez « Fournisseur » : « DELL » (en utilisant un « V » majuscule dans le fournisseur) dans un fichier, puis vous spécifiez « fournisseur » : Dans un autre fichier, le système renvoie une erreur si vous utilisez un « DELL » (en minuscules « v » dans le fournisseur).

> **Note**
>
> Vous devez enregistrer le fichier avec une extension .json et l'inventaire que vous définissez doit être composé uniquement de valeurs de chaîne.

Une fois le fichier créé, vous devez l'enregistrer sur l'instance. Le tableau suivant montre l'emplacement où les fichiers JSON d'inventaires personnalisés doivent être stockés sur l'instance :

Système d'exploitation	Path
Windows	%SystemDrive%\ProgramData\Amazon\SSM\InstanceData\<instance-id>\inventory\custom
Linux	/var/lib/amazon/ssm/<ID_instance>/inventory/custom

Pour obtenir un exemple d'utilisation de l'inventaire personnalisé, consultez Obtention de l'utilisation des disques de votre flotte à l'aide des types d'inventaires personnalisés EC2 Systems Manager.

Suppression de l'inventaire personnalisé

Vous pouvez utiliser l'action DeleteInventory d'API pour supprimer un type d'inventaire personnalisé et les données associées à ce type. Vous appelez la commande delete-inventory à l'aide de l'AWS CLI pour supprimer toutes les données d'un type d'inventaire. Vous appelez la commande delete-inventory avec l'option `SchemaDeleteOption` pour supprimer un type d'inventaire personnalisé.

> **Note**
>
> Un type d'inventaire est également appelé un schéma d'inventaire.

Le paramètre `SchemaDeleteOption` inclut les options suivantes :

- DeleteSchema: Cette option supprime le type personnalisé spécifié et toutes les données qui lui sont associées. Vous pouvez recréer le schéma plus tard, si vous le souhaitez.
- DisableSchema: Si vous choisissez cette option, le système désactive la version actuelle, supprime toutes les données correspondantes et ignore toutes les nouvelles données si la version est inférieure ou égale à la version désactivée. Vous pouvez réactiver ce type d'inventaire en appelant l'PutInventoryaction pour une version ultérieure à la version désactivée.

Pour supprimer ou désactiver l'inventaire personnalisé à l'aide de l'AWS CLI

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Exécutez la commande suivante pour utiliser l'option `dry-run` afin de voir quelles données seront supprimées du système. Cette commande ne supprime aucune donnée.

```
aws ssm delete-inventory --type-name "Custom:custom_type_name" --dry-run
```

   Le système renvoie des informations similaires à celles qui suivent.

```
{
   "DeletionSummary":{
      "RemainingCount":3,
      "SummaryItems":[
         {
            "Count":2,
            "RemainingCount":2,
            "Version":"1.0"
         },
         {
            "Count":1,
            "RemainingCount":1,
            "Version":"2.0"
         }
      ],
      "TotalCount":3
   },
   "TypeName":"Custom:custom_type_name"
}
```

   Pour plus d'informations sur la synthèse de la suppression d'inventaire, consultez Comprendre la synthèse de la suppression d'inventaire (p. 832).

3. Exécutez la commande suivante pour supprimer toutes les données d'un type inventaire personnalisé.

```
aws ssm delete-inventory --type-name "Custom:custom_type_name"
```

   > **Note**
   >
   > La sortie de cette commande n'affiche pas la progression de la suppression. Pour cette raison, TotalCount et le nombre restant sont toujours les mêmes, car le système n'a encore

rien supprimé. Vous pouvez utiliser la commande describe-inventory-deletions pour afficher la progression de la suppression, comme décrit plus loin dans cette rubrique.

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "DeletionId":"system_generated_deletion_ID",
   "DeletionSummary":{
      "RemainingCount":3,
      "SummaryItems":[
         {
            "Count":2,
            "RemainingCount":2,
            "Version":"1.0"
         },
         {
            "Count":1,
            "RemainingCount":1,
            "Version":"2.0"
         }
      ],
      "TotalCount":3
   },
   "TypeName":"custom_type_name"
}
```

Le système supprime toutes les données du type d'inventaire personnalisé spécifié du service d'inventaire Systems Manager.

4. Exécutez la commande suivante . La commande effectue les actions suivantes pour la version actuelle du type d'inventaire : elle désactive la version actuelle, supprime toutes les données de celle-ci et ignore toutes les nouvelles données si la version est antérieure ou égale à la version désactivée.

```
aws ssm delete-inventory --type-name "Custom:custom_type_name" --schema-delete-option
 "DisableSchema"
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "DeletionId":"system_generated_deletion_ID",
   "DeletionSummary":{
      "RemainingCount":3,
      "SummaryItems":[
         {
            "Count":2,
            "RemainingCount":2,
            "Version":"1.0"
         },
         {
            "Count":1,
            "RemainingCount":1,
            "Version":"2.0"
         }
      ],
      "TotalCount":3
   },
   "TypeName":"Custom:custom_type_name"
}
```

Vous pouvez afficher un type d'inventaire désactivé à l'aide de la commande suivante.

```
aws ssm get-inventory-schema --type-name Custom:custom_type_name
```

5. Exécutez la commande suivante pour supprimer un type d'inventaire.

```
aws ssm delete-inventory --type-name "Custom:custom_type_name" --schema-delete-option
 "DeleteSchema"
```

Le système supprime le schéma et toutes les données d'inventaire du type personnalisé spécifié.

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "DeletionId":"system_generated_deletion_ID",
   "DeletionSummary":{
      "RemainingCount":3,
      "SummaryItems":[
         {
            "Count":2,
            "RemainingCount":2,
            "Version":"1.0"
         },
         {
            "Count":1,
            "RemainingCount":1,
            "Version":"2.0"
         }
      ],
      "TotalCount":3
   },
   "TypeName":"Custom:custom_type_name"
}
```

Affichage du statut de suppression

Vous pouvez consulter le statut d'une opération de suppression à l'aide de la commande de l'AWS CLI describe-inventory-deletions. Vous pouvez spécifier un ID de suppression pour afficher le statut d'une opération de suppression spécifique. Sinon, vous pouvez omettre l'ID de suppression pour afficher la liste de toutes les suppressions exécutées au cours des 30 derniers jours.

1. Exécutez la commande suivante pour afficher le statut d'une opération de suppression. Le système a renvoyé l'ID de suppression dans la synthèse de suppression d'inventaire.

```
aws ssm describe-inventory-deletions --deletion-id system_generated_deletion_ID
```

Le système renvoie la statut le plus récent. L'opération de suppression peut ne pas être encore terminée. Le système renvoie des informations similaires à celles qui suivent.

```
{"InventoryDeletions":
  [
    {"DeletionId": "system_generated_deletion_ID",
     "DeletionStartTime": 1521744844,
     "DeletionSummary":
      {"RemainingCount": 1,
       "SummaryItems":
        [
          {"Count": 1,
```

```
            "RemainingCount": 1,
            "Version": "1.0"}
        ],
       "TotalCount": 1},
     "LastStatus": "InProgress",
     "LastStatusMessage": "The Delete is in progress",
     "LastStatusUpdateTime": 1521744844,
     "TypeName": "Custom:custom_type_name"}
  ]
}
```

Si l'opération de suppression aboutit, le `LastStatusMessage` indique : La suppression a réussi.

```
{"InventoryDeletions":
  [
    {"DeletionId": "system_generated_deletion_ID",
     "DeletionStartTime": 1521744844,
     "DeletionSummary":
      {"RemainingCount": 0,
       "SummaryItems":
        [
          {"Count": 1,
           "RemainingCount": 0,
           "Version": "1.0"}
        ],
       "TotalCount": 1},
     "LastStatus": "Complete",
     "LastStatusMessage": "Deletion is successful",
     "LastStatusUpdateTime": 1521745253,
     "TypeName": "Custom:custom_type_name"}
  ]
}
```

2. Exécutez la commande suivante pour afficher la liste de toutes les suppressions exécutées au cours des 30 derniers jours.

```
aws ssm describe-inventory-deletions --max-results a number
```

```
{"InventoryDeletions":
  [
    {"DeletionId": "system_generated_deletion_ID",
     "DeletionStartTime": 1521682552,
     "DeletionSummary":
      {"RemainingCount": 0,
       "SummaryItems":
        [
          {"Count": 1,
           "RemainingCount": 0,
           "Version": "1.0"}
        ],
       "TotalCount": 1},
     "LastStatus": "Complete",
     "LastStatusMessage": "Deletion is successful",
     "LastStatusUpdateTime": 1521682852,
     "TypeName": "Custom:custom_type_name"},
    {"DeletionId": "system_generated_deletion_ID",
     "DeletionStartTime": 1521744844,
     "DeletionSummary":
      {"RemainingCount": 0,
       "SummaryItems":
        [
```

```
          {"Count": 1,
           "RemainingCount": 0,
           "Version": "1.0"}
        ],
      "TotalCount": 1},
    "LastStatus": "Complete",
    "LastStatusMessage": "Deletion is successful",
    "LastStatusUpdateTime": 1521745253,
    "TypeName": "Custom:custom_type_name"},
   {"DeletionId": "system_generated_deletion_ID",
    "DeletionStartTime": 1521680145,
    "DeletionSummary":
     {"RemainingCount": 0,
      "SummaryItems":
       [
         {"Count": 1,
          "RemainingCount": 0,
          "Version": "1.0"}
        ],
      "TotalCount": 1},
    "LastStatus": "Complete",
    "LastStatusMessage": "Deletion is successful",
    "LastStatusUpdateTime": 1521680471,
    "TypeName": "Custom:custom_type_name"}
  ],
 "NextToken": "next-token"
```

Comprendre la synthèse de la suppression d'inventaire

Pour vous aider à comprendre le contenu de la synthèse de la suppression d'inventaire, prenez l'exemple suivant. Un utilisateur a attribué l'inventaire Custom:RackSpace à trois instances. Les éléments d'inventaire 1 et 2 utilisent le type personnalisé version 1.0 (« SchemaVersion » : « 1.0 »). L'élément d'inventaire 3 utilise le type personnalisé version 2.0 (« SchemaVersion » : « 2.0 »).

inventaire RackSpace personnalisé 1

```
{
   "CaptureTime":"2018-02-19T10:48:55Z",
   "TypeName":"CustomType:RackSpace",
   "InstanceId":"i-1234567890",
   "SchemaVersion":"1.0"   "Content":[
      {
         content of custom type omitted
      }
   ]
}
```

inventaire RackSpace personnalisé 2

```
{
   "CaptureTime":"2018-02-19T10:48:55Z",
   "TypeName":"CustomType:RackSpace",
   "InstanceId":"i-1234567891",
   "SchemaVersion":"1.0"   "Content":[
      {
         content of custom type omitted
      }
   ]
}
```

inventaire RackSpace personnalisé 3

```
{
   "CaptureTime":"2018-02-19T10:48:55Z",
   "TypeName":"CustomType:RackSpace",
   "InstanceId":"i-1234567892",
   "SchemaVersion":"2.0"   "Content":[
      {
         content of custom type omitted
      }
   ]
}
```

L'utilisateur exécute la commande suivante pour afficher un aperçu des données qui seront supprimées.

```
aws ssm delete-inventory --type-name "Custom:RackSpace" --dry-run
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "DeletionId":"1111-2222-333-444-66666",
   "DeletionSummary":{
      "RemainingCount":3,
      "TotalCount":3,
                TotalCount and RemainingCount are the number of items that would be deleted
 if this was not a dry run. These numbers are the same because the system didn't delete
 anything.
      "SummaryItems":[
         {
            "Count":2,             The system found two items that use SchemaVersion 1.0.
 Neither item was deleted.
            "RemainingCount":2,
            "Version":"1.0"
         },
         {
            "Count":1,             The system found one item that uses SchemaVersion 1.0.
 This item was not deleted.
            "RemainingCount":1,
            "Version":"2.0"
         }
      ],

   },
   "TypeName":"Custom:RackSpace"
}
```

L'utilisateur exécute la commande suivante pour supprimer l'inventaire Custom:RackSpace.

> **Note**
>
> La sortie de cette commande n'affiche pas la progression de la suppression. Pour cette raison, TotalCount et le nombre restant sont toujours les mêmes, car le système n'a encore rien supprimé. Vous pouvez utiliser la commande describe-inventory-deletions pour afficher la progression de la suppression.

```
aws ssm delete-inventory --type-name "Custom:RackSpace"
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "DeletionId":"1111-2222-333-444-7777777",
   "DeletionSummary":{
      "RemainingCount":3,           There are three items to delete
```

```
      "SummaryItems":[
         {
            "Count":2,              The system found two items that use SchemaVersion 1.0.
            "RemainingCount":2,
            "Version":"1.0"
         },
         {
            "Count":1,              The system found one item that uses SchemaVersion 2.0.
            "RemainingCount":1,
            "Version":"2.0"
         }
      ],
      "TotalCount":3
   },
   "TypeName":"RackSpace"
}
```

Affichage des actions de suppression d'inventaire dans EventBridge

Vous pouvez configurer Amazon EventBridge pour créer un événement à chaque fois qu'un utilisateur supprime l'inventaire personnalisé. EventBridge propose trois types d'événements pour les opérations de suppression d'inventaire personnalisé :

- Action de suppression pour une instance : Si l'inventaire personnalisé pour une instance gérée spécifique a été correctement supprimé ou non.
- Récapitulatif des actions de suppression : Résumé de l'action de suppression.
- Avertissement pour le type d'inventaire personnalisé désactivé : Événement d'avertissement si un utilisateur a appelé l'action PutInventory d'API pour une version de type d'inventaire personnalisé qui a été précédemment désactivée.

Voici des exemples de chaque événement :

Action de suppression pour une instance

```
{
   "version":"0",
   "id":"998c9cde-56c0-b38b-707f-0411b3ff9d11",
   "detail-type":"Inventory Resource State Change",
   "source":"aws.ssm",
   "account":"478678815555",
   "time":"2018-05-24T22:24:34Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:ssm:us-east-1:478678815555:managed-instance/i-0a5feb270fc3f0b97"
   ],
   "detail":{
      "action-status":"succeeded",
      "action":"delete",
      "resource-type":"managed-instance",
      "resource-id":"i-0a5feb270fc3f0b97",
      "action-reason":"",
      "type-name":"Custom:MyInfo"
   }
}
```

Récapitulatif d'action de suppression

```
{
   "version":"0",
   "id":"83898300-f576-5181-7a67-fb3e45e4fad4",
   "detail-type":"Inventory Resource State Change",
```

```
   "source":"aws.ssm",
   "account":"478678815555",
   "time":"2018-05-24T22:28:25Z",
   "region":"us-east-1",
   "resources":[

   ],
   "detail":{
      "action-status":"succeeded",
      "action":"delete-summary",
      "resource-type":"managed-instance",
      "resource-id":"",
      "action-reason":"The delete for type name Custom:MyInfo was completed. The deletion
 summary is: {\"totalCount\":2,\"remainingCount\":0,\"summaryItems\":[{\"version\":\"1.0\",
\"count\":2,\"remainingCount\":0}]}",
      "type-name":"Custom:MyInfo"
   }
}
```

Avertissement pour le type d'inventaire personnalisé désactivé

```
{
   "version":"0",
   "id":"49c1855c-9c57-b5d7-8518-b64aeeef5e4a",
   "detail-type":"Inventory Resource State Change",
   "source":"aws.ssm",
   "account":"478678815555",
   "time":"2018-05-24T22:46:58Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:ssm:us-east-1:478678815555:managed-instance/i-0ee2d86a2cfc371f6"
   ],
   "detail":{
      "action-status":"failed",
      "action":"put",
      "resource-type":"managed-instance",
      "resource-id":"i-0ee2d86a2cfc371f6",
      "action-reason":"The inventory item with type name Custom:MyInfo was sent with a
 disabled schema version 1.0. You must send a version greater than 1.0",
      "type-name":"Custom:MyInfo"
   }
}
```

Utilisez la procédure suivante pour créer une EventBridge règle pour les opérations de suppression d'inventaire personnalisé. Cette procédure vous montre comment créer une règle qui envoie des notifications à une rubrique Amazon SNS pour les opérations de suppression d'inventaire personnalisé. Avant de commencer, vérifiez que vous disposez d'une rubrique Amazon SNS, ou créez-en une nouvelle. Pour plus d'informations, consultez la section Getting Started du manuel Amazon Simple Notification Service Manuel du développeur.

Pour configurer EventBridge pour les opérations de suppression d'inventaire

1. Ouvrez la console Amazon EventBridge à l'adresse https://console.aws.amazon.com/events/.
2. Dans le volet de navigation, choisissez Rules (Règles), puis Create rule (Créer une règle).

   -ou-

   Si la page Amazon EventBridge d'accueil s'ouvre d'abord, choisissez Create rule (Créer une règle).
3. Saisissez un nom et une description pour la règle.

   Une règle ne peut pas avoir le même nom qu'une autre règle de la même région et sur le même bus d'événement.

4. Pour Define pattern (Définir un modèle), choisissez Event pattern (Modèle d'événement).
5. Choisissez Pré-defined pattern by service (Modèle prédéfini par service).
6. Pour Service provider (Fournisseur de services), choisissez AWS.
7. Pour Service name (Nom du service), choisissez EC2 Simple (SSM)Systems Manager
8. Pour Event type (Type d'événement), choisissez Inventory (Inventaire).
9. Vérifiez qu'un type de détail est sélectionné.
10. Pour Select event bus (Sélectionner un bus d'événement), choisissez le bus d'événement que vous souhaitez associer à cette règle. Si vous souhaitez que cette règle se déclenche sur les événements correspondants provenant de votre propre compte AWS, sélectionnez AWS default event bus (Bus d'événement par défaut AWS). Lorsqu'un service AWS de votre compte émet un événement, il accède toujours au bus d'événement par défaut de votre compte.
11. Pour Target (Cible), choisissez SNS topic (Rubrique SNS), puis sélectionnez votre rubrique dans la liste Topic (Rubrique).
12. Développez Configure input (Configurer l'entrée) et vérifiez que l'événement Matched (Correspondance) est sélectionné.
13. (Facultatif) Entrez une ou plusieurs balises pour la règle. Pour de plus amples informations, veuillez consulter Balisage de vos ressources Amazon EventBridge dans le Guide de l'utilisateur Amazon EventBridge.
14. Sélectionnez Create.

Affichage de l'historique d'inventaire et suivi des modifications

Vous pouvez consulter l'historique d'inventaire Systems Manager et le suivi des modifications pour l'ensemble de vos instances gérées avecAWS Config. AWS Config fournit une vue détaillée de la configuration des ressources AWS dans votre compte AWS. Elle indique comment les ressources sont liées entre elles et comment elles ont été configurées dans le passé, pour que vous puissiez observer comment les configurations et les relations changent au fil du temps. Pour afficher l'historique d'inventaire et le suivi des modifications, vous devez activer les ressources suivantes dans AWS Config.

- SSM:ManagedInstanceInventory
- SSM:PatchCompliance
- SSM:AssociationCompliance
- Données de fichier SSM

> **Note**
>
> En activant SSM:PatchCompliance et SSM:AssociationCompliance, vous pouvez afficher le suivi des modifications et l'historique des associations et des Gestionnaire de correctifs correctifs.State Manager Pour plus d'informations sur la gestion de la conformité pour ces ressources, consultez Utilisation de la conformité de configuration (p. 782).

La procédure suivante explique comment activer l'enregistrement de l'historique d'inventaire et du suivi des modifications dans AWS Config à l'aide de l'AWS CLI. Pour plus d'informations sur le choix et la configuration de ces ressources dans AWS Config, consultez Sélection des ressources enregistrées par AWS Config dans le AWS Config Developer Guide. Pour plus d'informations sur la tarification AWS Config, consultez Tarification .

Avant de commencer

AWS Config nécessite des autorisations AWS Identity and Access Management (IAM) pour obtenir les détails de configuration relatifs aux ressources Systems Manager. Dans la procédure suivante, vous

devez spécifier un nom Amazon Resource Name (ARN) pour un rôle IAM qui accorde à AWS Config une autorisation sur les ressources Systems Manager. Vous pouvez attacher la stratégie AWSConfigRole gérée au IAM rôle que vous attribuez à AWS Config. Pour plus d'informations sur la façon de créer un IAM rôle et d'attribuer la stratégie AWSConfigRole gérée à ce rôle, consultez Création d'un rôle pour déléguer des autorisations à un service AWS dans le IAM Guide de l'utilisateur .

Pour activer l'enregistrement de l'historique d'inventaire et du suivi des modifications dans AWS Config

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).
2. Copiez et collez l'exemple JSON suivant dans un fichier texte simple et enregistrez-le sous recordingGroup le nom .json.

```
{
   "allSupported":false,
   "includeGlobalResourceTypes":false,
   "resourceTypes":[
      "AWS::SSM::AssociationCompliance",
      "AWS::SSM::PatchCompliance",
      "AWS::SSM::ManagedInstanceInventory",
      "AWS::SSM::FileData"
   ]
}
```

3. Exécutez la commande suivante pour charger le fichier recordingGroup .json dans AWS Config .

```
aws configservice put-configuration-recorder --configuration-recorder
 name=myRecorder,roleARN=arn:aws:iam::123456789012:role/myConfigRole --recording-group
 file://recordingGroup.json
```

4. Exécutez la commande suivante pour commencer l'enregistrement de l'historique d'inventaire et du suivi des modifications.

```
aws configservice start-configuration-recorder --configuration-recorder-name myRecorder
```

Une fois que vous avez configuré l'historique et le suivi des modifications, vous pouvez explorer en détail l'historique pour rechercher une instance gérée spécifique en choisissant le bouton AWS Config dans la console Systems Manager.

Vous pouvez accéder au bouton AWS Config à partir de la page Instances gérées ou de la page Inventaire. En fonction de la taille de votre moniteur, vous devrez peut-être faire défiler l'affichage vers le côté droit de la page pour voir le bouton.

Procédures pas à pas de l'inventaire Systems Manager

Utilisez les procédures pas à pas suivantes pour collecter et gérer les données d'inventaire. Nous vous recommandons de réaliser ces procédures pas à pas avec les instances gérées dans un environnement de test au préalable.

Avant de commencer

Avant de commencer ces procédures, exécutez les tâches suivantes.

- Mettez à jour Agent SSM sur les instances à inventorier. En exécutant la dernière version de Agent SSM, vous êtes sûr de collecter les métadonnées de tous les types d'inventaire pris en charge. Pour plus d'informations sur la mise à jour de l'Agent SSM à l'aide de State Manager, consultez Mise à jour automatique d'Agent SSM (interface de ligne de commande) (p. 1043).
- Vérifiez que vos instances respectent les prérequis de Systems Manager. Pour plus d'informations, consultez la section Prérequis d'Systems Manager (p. 15).
- (Facultatif) Créez un fichier JSON pour collecter l'inventaire personnalisé. Pour plus d'informations, consultez la section Utilisation de l'inventaire personnalisé (p. 826).

Table des matières

- Procédure : Attribuer des métadonnées d'inventaire personnalisé à une instance (p. 838)
- Procédure : Configurer vos instances gérées pour l'inventaire à l'aide de l'interface de ligne de commande (p. 839)
- Procédure : Utiliser la synchronisation de données de ressources pour regrouper les données d'inventaire (p. 843)

Procédure : Attribuer des métadonnées d'inventaire personnalisé à une instance

La procédure suivante vous guide à travers le processus d'utilisation de lPutInventory'action d'API pour attribuer des métadonnées d'inventaire personnalisé à une instance gérée. Cet exemple attribue les informations sur les emplacements des racks à une instance. Pour plus d'informations sur l'inventaire personnalisé, consultez Utilisation de l'inventaire personnalisé (p. 826).

Pour attribuer des métadonnées d'inventaire personnalisé à une instance

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).
2. Exécutez la commande suivante pour attribuer les informations de localisation de rack à une instance.

```
aws ssm put-inventory --instance-id "ID" --items '[{"CaptureTime":
 "2016-08-22T10:01:01Z", "TypeName": "Custom:RackInfo", "Content":[{"RackLocation":
 "Bay B/Row C/Rack D/Shelf E"}], "SchemaVersion": "1.0"}]'
```

3. Exécutez la commande suivante pour afficher les entrées de l'inventaire personnalisé de cette instance.

```
aws ssm list-inventory-entries --instance-id ID --type-name "Custom:RackInfo"
```

   Le système répond en renvoyant des informations similaires à celles qui suivent.

```
{
    "InstanceId": "ID",
    "TypeName": "Custom:RackInfo",
    "Entries": [
        {
            "RackLocation": "Bay B/Row C/Rack D/Shelf E"
        }
    ],
    "SchemaVersion": "1.0",
    "CaptureTime": "2016-08-22T10:01:01Z"
}
```

4. Exécutez la commande suivante pour afficher le schéma d'inventaire personnalisé.

```
aws ssm get-inventory-schema --type-name Custom:RackInfo
```

Le système répond en renvoyant des informations similaires à celles qui suivent.

```
{
    "Schemas": [
        {
            "TypeName": "Custom:RackInfo",
            "Version": "1.0",
            "Attributes": [
                {
                    "DataType": "STRING",
                    "Name": "RackLocation"
                }
            ]
        }
    ]
}
```

Procédure : Configurer vos instances gérées pour l'inventaire à l'aide de l'interface de ligne de commande

Les procédures suivantes vous guident tout au long du processus de configuration de l'inventaire Systems Manager pour collecter les métadonnées de vos instances gérées. Lorsque vous configurez la collecte d'inventaire, vous commencez par créer une association Systems Manager State Manager. Systems Manager collecte les données d'inventaire lorsque cette association est exécutée. Si vous ne créez pas l'association en premier et essayez d'appeler le plug-in aws:softwareInventory en utilisant, par exemple, la fonctionnalité Fonctionnalité Exécuter la commande, le système renvoie l'erreur suivante :

```
The aws:softwareInventory plugin can only be invoked via ssm-associate.
```

> **Note**
>
> Une instance ne peut avoir qu'une association d'inventaire configurée en même temps. Si vous configurez une instance avec deux associations d'inventaire ou plus, l'association n'est pas exécutée et aucune donnée d'inventaire n'est collectée.

Configuration rapide de toutes vos instances gérées pour l'inventaire (interface de ligne de commande)

Vous pouvez configurer rapidement toutes les instances gérées dans votre compte AWS et dans la région actuelle pour collecter les données d'inventaire. C'est ce qu'on appelle la création d'une association d'inventaire global. Pour créer une association d'inventaire global à l'aide de l'AWS CLI, utilisez un caractère générique comme valeur de `instanceIds`, comme illustré dans la procédure suivante.

Pour configurer l'inventaire pour toutes les instances gérées dans votre compte AWS et dans la région actuelle (interface de ligne de commande)

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Exécutez la commande suivante .

Linux

```
aws ssm create-association \
--name AWS-GatherSoftwareInventory \
--targets Key=InstanceIds,Values=* \
--schedule-expression "rate(1 day)" \
--parameters
 applications=Enabled,awsComponents=Enabled,customInventory=Enabled,instanceDetailedInformation
```

Windows

```
aws ssm create-association ^
--name AWS-GatherSoftwareInventory ^
--targets Key=InstanceIds,Values=* ^
--schedule-expression "rate(1 day)" ^
--parameters
 applications=Enabled,awsComponents=Enabled,customInventory=Enabled,instanceDetailedInformation
```

Note

Cette commande ne permet pas à l'inventaire de collecter des métadonnées pour le registre ou les fichiers Windows. Pour effectuer l'inventaire de ces types de données, utilisez la procédure suivante.

Configuration manuelle de l'inventaire sur vos instances gérées (interface de ligne de commande)

Utilisez la procédure suivante pour configurer manuellement Systems Manager l'inventaire sur vos instances gérées à l'aide d'une instance IDs ou de balises.

Pour configurer manuellement vos instances gérées pour l'inventaire (interface de ligne de commande)

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Exécutez la commande suivante pour créer une association State Manager qui exécute l'inventaire Systems Manager sur l'instance. Cette commande configure le service pour qu'il s'exécute toutes les six heures et qu'il collecte les métadonnées de configuration du réseau, de Windows Update et des applications à partir d'une instance.

   Linux

   ```
   aws ssm create-association \
   --name "AWS-GatherSoftwareInventory" \
   --targets "Key=instanceids,Values=an_instance_ID" \
   --schedule-expression "rate(240 minutes)" \
   --output-location "{ \"S3Location\": { \"OutputS3Region\": \"region_ID, for example
    us-east-2\", \"OutputS3BucketName\": \"DOC-EXAMPLE-BUCKET\", \"OutputS3KeyPrefix
   \": \"Test\" } }" \
   --parameters "networkConfig=Enabled,windowsUpdates=Enabled,applications=Enabled"
   ```

   Windows

   ```
   aws ssm create-association ^
   ```

```
--name "AWS-GatherSoftwareInventory" ^
--targets "Key=instanceids,Values=an_instance_ID" ^
--schedule-expression "rate(240 minutes)" ^
--output-location "{ \"S3Location\": { \"OutputS3Region\": \"region_ID, for example
 us-east-2\", \"OutputS3BucketName\": \"DOC-EXAMPLE-BUCKET\", \"OutputS3KeyPrefix
\": \"Test\" } }" ^
--parameters "networkConfig=Enabled,windowsUpdates=Enabled,applications=Enabled"
```

Le système répond en renvoyant des informations similaires à celles qui suivent.

```
{
    "AssociationDescription": {
        "ScheduleExpression": "rate(240 minutes)",
        "OutputLocation": {
            "S3Location": {
                "OutputS3KeyPrefix": "Test",
                "OutputS3BucketName": "Test bucket",
                "OutputS3Region": "us-east-2"
            }
        },
        "Name": "The name you specified",
        "Parameters": {
            "applications": [
                "Enabled"
            ],
            "networkConfig": [
                "Enabled"
            ],
            "windowsUpdates": [
                "Enabled"
            ]
        },
        "Overview": {
            "Status": "Pending",
            "DetailedStatus": "Creating"
        },
        "AssociationId": "1a2b3c4d5e6f7g-1a2b3c-1a2b3c-1a2b3c-1a2b3c4d5e6f7g",
        "DocumentVersion": "$DEFAULT",
        "LastUpdateAssociationDate": 1480544990.06,
        "Date": 1480544990.06,
        "Targets": [
            {
                "Values": [
                   "i-02573cafcfEXAMPLE"
                ],
                "Key": "InstanceIds"
            }
        ]
    }
}
```

Vous pouvez cibler de grands groupes d'instances en utilisant le paramètre `Targets`avec les balises EC2. Exemple :

Linux

```
aws ssm create-association \
--name "AWS-GatherSoftwareInventory" \
--targets "Key=tag:Environment,Values=Production" \
--schedule-expression "rate(240 minutes)" \
```

```
--output-location "{ \"S3Location\": { \"OutputS3Region\": \"us-east-2\",
 \"OutputS3BucketName\": \"DOC-EXAMPLE-BUCKET\", \"OutputS3KeyPrefix\": \"Test
\" } }" \
--parameters "networkConfig=Enabled,windowsUpdates=Enabled,applications=Enabled"
```

Windows

```
aws ssm create-association ^
--name "AWS-GatherSoftwareInventory" ^
--targets "Key=tag:Environment,Values=Production" ^
--schedule-expression "rate(240 minutes)" ^
--output-location "{ \"S3Location\": { \"OutputS3Region\": \"us-east-2\",
 \"OutputS3BucketName\": \"DOC-EXAMPLE-BUCKET\", \"OutputS3KeyPrefix\": \"Test
\" } }" ^
--parameters "networkConfig=Enabled,windowsUpdates=Enabled,applications=Enabled"
```

Vous pouvez également inventorier des fichiers et des clés de registre Windows sur une instance Windows Server en utilisant les types d'inventaire `files` et `windowsRegistry` avec des expressions. Pour plus d'informations sur ces types d'inventaire, consultez Utilisation de l'inventaire de fichiers et du registre Windows (p. 798).

Linux

```
aws ssm create-association \
--name "AWS-GatherSoftwareInventory" \
--targets "Key=instanceids,Values=i-0704358e3a3da9eb1" \
--schedule-expression "rate(240 minutes)" \
--parameters '{"files":["[{\"Path\": \"C:\\Program Files\", \"Pattern\": [\"*.exe
\"], \"Recursive\": true}]"], "windowsRegistry": ["[{\"Path\":\"HKEY_LOCAL_MACHINE\
\Software\\Amazon\", \"Recursive\":true}]"]}' \
--profile dev-pdx
```

Windows

```
aws ssm create-association ^
--name "AWS-GatherSoftwareInventory" ^
--targets "Key=instanceids,Values=i-0704358e3a3da9eb1" ^
--schedule-expression "rate(240 minutes)" ^
--parameters '{"files":["[{\"Path\": \"C:\\Program Files\", \"Pattern\": [\"*.exe
\"], \"Recursive\": true}]"], "windowsRegistry": ["[{\"Path\":\"HKEY_LOCAL_MACHINE\
\Software\\Amazon\", \"Recursive\":true}]"]}' ^
--profile dev-pdx
```

3. Exécutez la commande suivante pour afficher le statut de l'association.

```
aws ssm describe-instance-associations-status --instance-id an_instance_ID
```

Le système répond en renvoyant des informations similaires à celles qui suivent.

```
{
"InstanceAssociationStatusInfos": [
         {
            "Status": "Pending",
            "DetailedStatus": "Associated",
            "Name": "reInvent2016PolicyDocumentTest",
            "InstanceId": "i-1a2b3c4d5e6f7g",
            "AssociationId": "1a2b3c4d5e6f7g-1a2b3c-1a2b3c-1a2b3c-1a2b3c4d5e6f7g",
            "DocumentVersion": "1"
```

```
        }
]
}
```

Procédure : Utiliser la synchronisation de données de ressources pour regrouper les données d'inventaire

Utilisez la procédure suivante pour créer une synchronisation de données de ressource pour l'inventaire en utilisant la console AWS CLI. Une synchronisation de données de ressources transfère automatiquement les données d'inventaire collectées depuis toutes vos instances gérées vers un compartiment S3 central. La synchronisation met automatiquement à jour les données dans le compartiment S3 central lors de la découverte de nouvelles données d'inventaire. Cette procédure pas à pas indique également comment utiliser Amazon Athena et Amazon QuickSight pour interroger et analyser les données agrégées. Pour de plus amples informations sur la création d'une synchronisation de données de ressource à l'aide de la console Amazon EC2, veuillez consulter Configuration de la synchronisation de données de ressource pour l'inventaire (p. 800).

> **Note**
>
> Cette procédure pas à pas comporte des informations sur la façon de chiffrer la synchronisation avec AWS Key Management Service (AWS KMS). Comme l'inventaire ne collecte aucune donnée spécifique à l'utilisateur, propriétaire ou sensible, le chiffrement est facultatif. Pour plus d'informations sur AWS KMS, consultez AWS Key Management Service Developer Guide.

Avant de commencer

Avant de démarrer cette procédure pas à pas, vous devez collecter les métadonnées d'inventaire depuis vos instances gérées. En vue des sections Amazon Athena et Amazon QuickSight de cette procédure pas à pas, nous vous recommandons de collecter les métadonnées d'application. Pour plus d'informations sur la façon de collecter des données d'inventaire, consultez Procédure : Configurer vos instances gérées pour l'inventaire à l'aide de l'interface de ligne de commande (p. 839).

(Facultatif) Si vous souhaitez chiffrer la synchronisation avec AWS KMS, vous devez créer une nouvelle clé incluant la stratégie suivante, ou mettre à jour une clé existante et lui ajouter cette stratégie.

```
{
   "Version":"2012-10-17",
   "Id":"ssm-access-policy",
   "Statement":[
      {
         "Sid":"ssm-access-policy-statement",
         "Action":[
            "kms:GenerateDataKey"
         ],
         "Effect":"Allow",
         "Principal":{
            "Service":"ssm.amazonaws.com"
         },
         "Resource":"arn:aws:kms:region:AWS-account-ID:key/KMS-key-id"
      }
   ]
}
```

Pour créer une synchronisation des données de ressource pour l'inventaire

1. Ouvrez la console Amazon S3 à l'adresse https://console.aws.amazon.com/s3/.
2. Créez un compartiment pour stocker vos données d'inventaire agrégées. Pour de plus amples informations, veuillez consulter Créer un compartiment dans le Amazon Simple Storage Service Guide de mise en route. Notez le nom du compartiment et la région AWS dans laquelle vous l'avez créé.

3. Après avoir créé le compartiment, sélectionnez l'onglet Autorisations, puis choisissez Stratégie de compartiment.
4. Copiez et collez la stratégie de compartiment suivante dans l'éditeur de stratégie. Remplacez *DOC-EXAMPLE-BUCKET* et *account-id* par le nom du compartiment S3 que vous avez créé et un ID de compte AWS valide. (Facultatif) Remplacez *bucket-prefix* par le nom d'un Amazon S3 préfixe (sous-répertoire). Si vous n'avez pas créé de préfixe, supprimez *bucket-prefix/* à partir de l'ARN de la stratégie.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "SSMBucketPermissionsCheck",
            "Effect": "Allow",
            "Principal": {
                "Service": "ssm.amazonaws.com"
            },
            "Action": "s3:GetBucketAcl",
            "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET"
        },
        {
            "Sid": " SSMBucketDelivery",
            "Effect": "Allow",
            "Principal": {
                "Service": "ssm.amazonaws.com"
            },
            "Action": "s3:PutObject",
            "Resource": ["arn:aws:s3:::DOC-EXAMPLE-BUCKET/bucket-prefix/*/
accountid=account-id/*"],
            "Condition": {
                "StringEquals": {
                    "s3:x-amz-acl": "bucket-owner-full-control"
                }
            }
        }
    ]
}
```

5. (Facultatif) Si vous voulez chiffrer la synchronisation, vous devez ajouter la stratégie suivante au compartiment. Répétez l'étape précédente pour ajouter la stratégie suivante au compartiment.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Principal":{
            "Service":"ssm.amazonaws.com"
         },
         "Action":"s3:PutObject",
         "Resource":"arn:aws:s3:::DOC-EXAMPLE-BUCKET/bucket-prefix/*/accountid=*/*",
         "Condition":{
            "StringEquals":{
               "s3:x-amz-server-side-encryption":"aws:kms",
               "s3:x-amz-server-side-encryption-aws-kms-key-
id":"arn:aws:kms:region:account_ID:key/KMS_key_ID"
            }
         }
      }
   ]
}
```

6. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

7. (Facultatif) Si vous voulez chiffrer la synchronisation, exécutez la commande suivante afin de vérifier que la stratégie de compartiment applique la clé KMS requise.

   Linux

   ```
   aws s3 cp ./A_file_in_the_bucket s3://DOC-EXAMPLE-BUCKET/prefix/ \
   --sse aws:kms \
   --sse-kms-key-id "arn:aws:kms:region:account_ID:key/KMS_key_id" \
   --region region, for example, us-east-2
   ```

   Windows

   ```
   aws s3 cp ./A_file_in_the_bucket s3://DOC-EXAMPLE-BUCKET/prefix/ ^
       --sse aws:kms ^
       --sse-kms-key-id "arn:aws:kms:region:account_ID:key/KMS_key_id" ^
       --region region, for example, us-east-2
   ```

8. Exécutez la commande suivante pour créer une configuration de synchronisation de données de ressource avec le compartiment S3 que vous avez créé au début de cette procédure. Cette commande crée une synchronisation à partir de la région AWS dans laquelle vous êtes actuellement connecté.

   > **Note**
   >
   > Si la synchronisation et le compartiment S3 cible sont localisés dans des régions différentes, vous pourriez être sujet à une tarification de transfert de données. Pour de plus amples informations, veuillez consulter Tarification Amazon S3.

   Linux

   ```
   aws ssm create-resource-data-sync \
   --sync-name a_name \
   --s3-destination "BucketName=DOC-EXAMPLE-BUCKET,Prefix=prefix_name,
    if_specified,SyncFormat=JsonSerDe,Region=bucket_region"
   ```

   Windows

   ```
   aws ssm create-resource-data-sync ^
   --sync-name a_name ^
   --s3-destination "BucketName=DOC-EXAMPLE-BUCKET,Prefix=prefix_name,
    if_specified,SyncFormat=JsonSerDe,Region=bucket_region"
   ```

   Vous pouvez utiliser le paramètre `region` pour spécifier l'emplacement dans lequel la configuration de synchronisation doit être créée. Dans l'exemple suivant, les données d'inventaire de la région us-west-1 seront synchronisées dans le compartiment S3 de la région us-west-2.

   Linux

   ```
   aws ssm create-resource-data-sync \
       --sync-name InventoryDataWest \
       --s3-destination "BucketName=DOC-EXAMPLE-
   BUCKET,Prefix=HybridEnv,SyncFormat=JsonSerDe,Region=us-west-2"
       --region us-west-1
   ```

Windows

```
aws ssm create-resource-data-sync ^
--sync-name InventoryDataWest ^
--s3-destination "BucketName=DOC-EXAMPLE-
BUCKET,Prefix=HybridEnv,SyncFormat=JsonSerDe,Region=us-west-2" ^ --region us-west-1
```

(Facultatif) Si vous voulez chiffrer la synchronisation avec AWS KMS, exécutez la commande suivante pour créer la synchronisation. Si vous chiffrez la synchronisation, la clé AWS KMS et le compartiment S3 doivent se trouver dans la même région.

Linux

```
aws ssm create-resource-data-sync \
--sync-name sync_name \
--s3-destination "BucketName=DOC-EXAMPLE-BUCKET,Prefix=prefix_name,
 if_specified,SyncFormat=JsonSerDe,AWSKMSKeyARN=arn:aws:kms:region:account_ID:key/
KMS_key_ID,Region=bucket_region" \
--region region
```

Windows

```
aws ssm create-resource-data-sync ^
--sync-name sync_name ^
--s3-destination "BucketName=DOC-EXAMPLE-BUCKET,Prefix=prefix_name,
 if_specified,SyncFormat=JsonSerDe,AWSKMSKeyARN=arn:aws:kms:region:account_ID:key/
KMS_key_ID,Region=bucket_region" ^
--region region
```

9. Exécutez la commande suivante pour afficher le statut de la configuration de synchronisation.

```
aws ssm list-resource-data-sync
```

Si vous avez créé la configuration de synchronisation dans une autre région, vous devez spécifier le paramètre `region`, comme illustré dans l'exemple suivant.

```
aws ssm list-resource-data-sync --region us-west-1
```

10. Une fois la configuration de synchronisation créée, recherchez le compartiment cible dans Amazon S3. Les données d'inventaire doivent apparaître en quelques minutes.

Utilisation des données dans Amazon Athena

La section suivante décrit comment afficher et interroger les données dans Amazon Athena. Avant de commencer, nous vous recommandons de vous familiariser avec Athena. Pour de plus amples informations, veuillez consulter Qu'est-ce-qu'Amazon Athena ? et Utilisation des données dans le Amazon Athena Guide de l'utilisateur.

Pour afficher et interroger les données dans Amazon Athena

1. Ouvrez la console Athena à l'adresse https://console.aws.amazon.com/athena/.
2. Copiez et collez la déclaration suivante dans l'éditeur de requête, puis choisissez Exécuter la requête.

```
CREATE DATABASE ssminventory
```

Le système crée une base de données nommée ssminventory.

3. Copiez et collez la déclaration suivante dans l'éditeur de requête, puis choisissez Exécuter la requête. Remplacez *DOC-EXAMPLE-BUCKET* et *bucket_prefix* par le nom et le préfixe de la Amazon S3 cible.

```
CREATE EXTERNAL TABLE IF NOT EXISTS ssminventory.AWS_Application (
Name string,
ApplicationType string,
Publisher string,
Version string,
InstalledTime string,
Architecture string,
URL string,
Summary string,
PackageId string
)
PARTITIONED BY (AccountId string, Region string, ResourceType string)
ROW FORMAT SERDE 'org.openx.data.jsonserde.JsonSerDe'
WITH SERDEPROPERTIES (
  'serialization.format' = '1'
) LOCATION 's3://DOC-EXAMPLE-BUCKET/bucket_prefix/AWS:Application/'
```

4. Copiez et collez la déclaration suivante dans l'éditeur de requête, puis choisissez Exécuter la requête.

```
MSCK REPAIR TABLE ssminventory.AWS_Application
```

Le système effectue le partitionnement de la table.

> **Note**
>
> Si vous créez des synchronisations de données de ressources à partir de régions ou de comptes AWS supplémentaires, vous devez exécuter à nouveau cette commande pour mettre à jour les partitions. Il est probable que vous deviez également mettre à jour votre stratégie de compartiment S3.

5. Pour prévisualiser vos données, choisissez l'icône Aperçu située à côté de la table Application AWS.

6. Copiez et collez la déclaration suivante dans l'éditeur de requête, puis choisissez Exécuter la requête.

```
SELECT a.name, a.version, count( a.version) frequency
from aws_application a where
a.name = 'aws-cfn-bootstrap'
group by a.name, a.version
order  by frequency desc
```

La requête renvoie un nombre de versions différentes d'aws-cfn-bootstrap, une application AWS présente sur les instances EC2 pour Linux et Windows Server.

7. Copiez et collez individuellement les instructions suivantes dans l'éditeur de requête, remplacez *DOC-EXAMPLE-BUCKET* et *bucket-prefix* avec les informations relatives à Amazon S3 , puis choisissez Run Query. Ces déclarations définissent des tables d'inventaire supplémentaires dans Athena.

```
CREATE EXTERNAL TABLE IF NOT EXISTS ssminventory.AWS_AWSComponent (
 `ResourceId` string,
  `Name` string,
  `ApplicationType` string,
```

```
  `Publisher` string,
  `Version` string,
  `InstalledTime` string,
  `Architecture` string,
  `URL` string
)
PARTITIONED BY (AccountId string, Region string, ResourceType string)
ROW FORMAT SERDE 'org.openx.data.jsonserde.JsonSerDe'
WITH SERDEPROPERTIES (
  'serialization.format' = '1'
) LOCATION 's3://DOC-EXAMPLE-BUCKET/bucket-prefix/AWS:AWSComponent/'
```

```
MSCK REPAIR TABLE ssminventory.AWS_AWSComponent
```

```
CREATE EXTERNAL TABLE IF NOT EXISTS ssminventory.AWS_WindowsUpdate (
  `ResourceId` string,
  `HotFixId` string,
  `Description` string,
  `InstalledTime` string,
  `InstalledBy` string
)
PARTITIONED BY (AccountId string, Region string, ResourceType string)
ROW FORMAT SERDE 'org.openx.data.jsonserde.JsonSerDe'
WITH SERDEPROPERTIES (
  'serialization.format' = '1'
) LOCATION 's3://DOC-EXAMPLE-BUCKET/bucket-prefix/AWS:WindowsUpdate/'
```

```
MSCK REPAIR TABLE ssminventory.AWS_WindowsUpdate
```

```
CREATE EXTERNAL TABLE IF NOT EXISTS ssminventory.AWS_InstanceInformation (
  `AgentType` string,
  `AgentVersion` string,
  `ComputerName` string,
  `IamRole` string,
  `InstanceId` string,
  `IpAddress` string,
  `PlatformName` string,
  `PlatformType` string,
  `PlatformVersion` string
)
PARTITIONED BY (AccountId string, Region string, ResourceType string)
ROW FORMAT SERDE 'org.openx.data.jsonserde.JsonSerDe'
WITH SERDEPROPERTIES (
  'serialization.format' = '1'
) LOCATION 's3://DOC-EXAMPLE-BUCKET/bucket-prefix/AWS:InstanceInformation/'
```

```
MSCK REPAIR TABLE ssminventory.AWS_InstanceInformation
```

```
CREATE EXTERNAL TABLE IF NOT EXISTS ssminventory.AWS_Network (
  `ResourceId` string,
  `Name` string,
  `SubnetMask` string,
  `Gateway` string,
  `DHCPServer` string,
  `DNSServer` string,
  `MacAddress` string,
  `IPV4` string,
  `IPV6` string
```

```
)
PARTITIONED BY (AccountId string, Region string, ResourceType string)
ROW FORMAT SERDE 'org.openx.data.jsonserde.JsonSerDe'
WITH SERDEPROPERTIES (
  'serialization.format' = '1'
) LOCATION 's3://DOC-EXAMPLE-BUCKET/bucket-prefix/AWS:Network/'
```

```
MSCK REPAIR TABLE ssminventory.AWS_Network
```

```
CREATE EXTERNAL TABLE IF NOT EXISTS ssminventory.AWS_PatchSummary (
  `ResourceId` string,
  `PatchGroup` string,
  `BaselineId` string,
  `SnapshotId` string,
  `OwnerInformation` string,
  `InstalledCount` int,
  `InstalledOtherCount` int,
  `NotApplicableCount` int,
  `MissingCount` int,
  `FailedCount` int,
  `OperationType` string,
  `OperationStartTime` string,
  `OperationEndTime` string
)
PARTITIONED BY (AccountId string, Region string, ResourceType string)
ROW FORMAT SERDE 'org.openx.data.jsonserde.JsonSerDe'
WITH SERDEPROPERTIES (
  'serialization.format' = '1'
) LOCATION 's3://DOC-EXAMPLE-BUCKET/bucket-prefix/AWS:PatchSummary/'
```

```
MSCK REPAIR TABLE ssminventory.AWS_PatchSummary
```

Utilisation des données dans Amazon QuickSight

La section suivante fournit une présentation des liens pour la création d'une visualisation dans Amazon QuickSight.

Pour concevoir une visualisation dans Amazon QuickSight

1. Inscrivez-vous à Amazon QuickSight et connectez-vous à la QuickSight console .
2. Créez un ensemble de données depuis l'application AWS et toute autre table que vous avez créée. Pour de plus amples informations, veuillez consulter Création d'un ensemble de données à partir de données Amazon Athena.
3. Joignez les tables. Par exemple, vous pouvez joindre la colonne `instanceid` depuis `AWS_InstanceInformation` car elle correspond à la colonne `resourceid` dans d'autres tables d'inventaire. Pour plus d'informations sur la jonction de tables, consultez Jonction de tables.
4. Créez une visualisation. Pour plus d'informations, consultez Utilisation des QuickSight représentations visuelles Amazon.

Dépannage des problèmes liés à l'inventaire Systems Manager

Cette rubrique inclut des informations sur la résolution d'erreurs ou de problèmes courants liés à l'inventaire Systems Manager.

Rubriques

- L'état d'exécution de l'inventaire ne quitte jamais l'état En attente (p. 850)
- L'exécution du document AWS-ListWindowsInventory échoue (p. 850)
- La console n'affiche pas les éléments Tableau de bord d'inventaire | Vue détaillée | onglets Paramètres (p. 850)
- UnsupportedAgent (p. 851)
- Skipped (p. 851)
- Failed (p. 851)

L'état d'exécution de l'inventaire ne quitte jamais l'état En attente

Si une exécution d'inventaire ne quitte jamais le `Pending` statut, cela signifie que l'instance ne dispose pas d'une autorisation suffisante pour exécuter Systems Manager l'inventaire. Vérifiez que le profil d'instance AWS Identity and Access Management (IAM) inclut la stratégie AmazonSSMManagedInstanceCore gérée. Pour plus d'informations sur la façon d'ajouter cette stratégie à un profil d'instance, consultez Tâche 2 : Ajouter des autorisations à un profil dSystems Manager'instance (console) (p. 38) .

> **Note**
>
> Au minimum, le profil d'instance doit disposer des autorisations IAM suivantes.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeAssociation",
                "ssm:ListAssociations",
                "ssm:ListInstanceAssociations",
                "ssm:PutInventory",
                "ssm:PutComplianceItems",
                "ssm:UpdateAssociationStatus",
                "ssm:UpdateInstanceAssociationStatus",
                "ssm:UpdateInstanceInformation",
                "ssm:GetDocument",
                "ssm:DescribeDocument"
            ],
            "Resource": "*"
        }
    ]
}
```

L'exécution du document AWS-ListWindowsInventory échoue

Le document AWS-ListWindowsInventory est obsolète. N'utilisez pas ce document pour collecter l'inventaire. Utilisez plutôt l'un des processus décrits dans Configuration de la collecte d'inventaire (p. 806).

La console n'affiche pas les éléments Tableau de bord d'inventaire | Vue détaillée | onglets Paramètres

La page Detailed Inventory View (Vue détaillée d'inventaire) est uniquement disponible dans les régions AWS qui proposent Amazon Athena. Si les onglets suivants ne sont pas affichés sur la page d'inventaire, cela signifie qu'Athena n'est pas disponible dans la région et que vous ne pouvez pas utiliser la vue détaillée pour interroger les données.

UnsupportedAgent

Si le statut détaillé d'une association d'inventaire indique UnsupportedAgent , et si le statut d'association indique Failed, cela signifie que la version d' Agent SSM sur l'instance n'est pas correcte. Pour créer une association d'inventaire globale (pour toutes les instances d'inventaire de votre compte AWS) par exemple, vous devez utiliser Agent SSM, version 2.0.790.0 ou version ultérieure. Vous pouvez afficher la version de l'agent en cours d'exécution sur chacune de vos instances sur la page Instances gérées dans la colonne Version d'agent. Pour plus d'informations sur la mise à jour de l'Agent SSM sur vos instances, consultez Mettre à jour l'Agent SSM à l'aide de Fonctionnalité Exécuter la commande (p. 948).

Skipped

Si le statut de l'association d'inventaire pour une instance indique Ignoré, cela signifie que vous avez créé une association d'inventaire global, mais qu'une association d'inventaire était déjà attribuée à l'instance ignorée. L'association d'inventaire global n'a pas été attribuée à cette instance, et aucun inventaire n'a été collecté par l'association d'inventaire global. Toutefois, l'instance continue de signaler des données d'inventaire lorsque l'association d'inventaire spécifique s'exécute.

Failed

Si le statut de l'association d'inventaire pour une instance indique Échec, cela peut signifier que plusieurs plusieurs associations d'inventaire sont attribuées à l'instance. Une seule association d'inventaire peut être attribuée à une instance à la fois. Une association d'inventaire utilise le document SSM AWS-GatherSoftwareInventory. Vous pouvez exécuter la commande suivante à l'aide de l'AWS CLI pour afficher la liste des associations pour une instance.

```
aws ssm describe-instance-associations-status --instance-id instance ID
```

Instances gérées AWS Systems Manager

Une instance gérée est une machine configurée pour AWS Systems Manager. Vous pouvez configurer des instances EC2 ou des machines sur site dans un environnement hybride en tant qu'instances gérées. Systems Manager prend en charge plusieurs distributions de Linux, notamment les appareils Raspberry Pi et Microsoft Windows Server.

> **Note**
>
> Dans l'AWS Management Console, toute machine dotée du préfixe « mi- » est une instance gérée de serveur sur site ou de machine virtuelle (VM).

AWS Systems Manager offre un niveau d'instances standard et un niveau d'instances avancées pour les serveurs et machines virtuelles de votre environnement hybride. Le niveau des instances standard vous permet d'enregistrer 1 000 serveurs ou machines virtuelles au maximum par compte AWS et par région AWS. Si vous avez besoin d'enregistrer plus de 1 000 serveurs ou machines virtuelles dans un seul compte et une seule région, utilisez le niveau des instances avancées. Vous pouvez créer autant d'instances que vous le souhaitez dans le niveau des instances avancées, mais toutes les instances configurées pour Systems Manager sont facturées en fonction de l'utilisation. Pour plus d'informations sur l'activation des instances avancées, consultez Activation du niveau d'instances avancées (p. 853). Pour de plus amples informations sur la tarification, veuillez consulter Tarification d'AWS Systems Manager.

Note

- Les instances avancées vous permettent également de vous connecter à vos machines hybrides en utilisant AWS Systems Manager Session Manager. Session Manager fournit l'accès au shell interactif à vos instances. Pour plus d'informations, consultez AWS Systems Manager Session Manager (p. 869).
- Le quota d'instances standard s'applique également aux instances EC2 qui utilisent une activation sur site de Systems Manager (ce qui n'est pas un scénario courant).
- L'application des correctifs aux applications Microsoft est disponible uniquement sur les instances EC2 et dans le niveau d'instances avancées. Pour corriger les applications Microsoft sur les serveurs et les machines virtuelles sur site, vous devez activer le niveau d'instances avancées. Pour plus d'informations, consultez À propos des correctifs d'applications sur Windows Server (p. 1108).

Si vous ne voyez pas vos instances gérées répertoriées dans la console, procédez comme suit :

1. Vérifiez que la console est ouverte dans la AWS région dans laquelle vous avez créé vos instances gérées. Vous pouvez changer de région à l'aide de la liste figurant dans la partie supérieure droite de la console.
2. Vérifiez que vos instances respectent les exigences de Systems Manager. Pour plus d'informations, consultez la section Prérequis d'Systems Manager (p. 15).
3. Pour les serveurs et VMs dans un environnement hybride, vérifiez que vous avez terminé le processus d'activation. Pour plus d'informations, consultez la section Configuration d'AWS Systems Manager pour les environnements hybrides (p. 49).

Note

Systems Manager nécessite des références horaires précises pour pouvoir effectuer ses opérations. Si la date et l'heure de votre instance ne sont pas correctement définies, il est possible que la date de la signature ne corresponde pas à vos demandes d'API. Pour plus d'informations, consultez la section Cas d'utilisation et bonnes pratiques (p. 1403).

Vérification de la prise en charge de Systems Manager sur une instance

AWS Configfournit des règles AWS gérées prédéfinies et personnalisables que AWS Config utilise pour évaluer si vos configurations de AWS ressources sont conformes aux bonnes pratiques courantes.AWS Config Les règles gérées incluent la règle ec2-instance-managed-by-systems-manager. Cette règle vérifie si les instances EC2 de votre compte sont gérées par Systems Manager. Pour de plus amples informations, veuillez consulter Règles gérées AWS Config.

Vérifiez les prérequis Systems Manager

Pour plus d'informations sur les prérequis pour Systems Manager, consultez Prérequis d'Systems Manager (p. 15). Pour plus d'informations sur la configuration de serveurs sur site et VMs en tant qu'instances gérées, consultez Configuration d'AWS Systems Manager pour les environnements hybrides (p. 49) .

Renforcement de la sécurité sur les instances gérées

Pour de plus amples informations sur la manière de renforcer votre sécurité contre des commandes de niveau racine non autorisées sur vos instances, veuillez consulter la page Restriction de l'accès aux commandes de niveau racine viaAgent SSM (p. 108).

Rubriques

- Configuration des niveaux d'instance (p. 853)
- Réinitialisation de mots de passe sur des instances gérées (p. 860)
- Annuler l'enregistrement des instances gérées dans un environnement hybride (p. 864)
- Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865)

Configuration des niveaux d'instance

AWS Systems Managerpropose un niveau d'instances standard et un niveau d'instances avancées pour les serveurs et VMs dans votre environnement hybride. Le niveau d'instances standard vous permet d'enregistrer 1 000 serveurs sur site au maximum ou VMs par AWS compte et par AWS région. Si vous devez enregistrer plus de 1 000 serveurs sur site ou VMs dans un seul compte et une seule région, utilisez le niveau des instances avancées. Dans un environnement hybride, vous pouvez activer autant d'instances gérées que vous le souhaitez dans le niveau d'instances avancées. Toutefois, toutes les instances configurées pour Systems Manager à l'aide du processus d'activation d'instance gérée décrit dans Créer une activation d'instance gérée pour un environnement hybride (p. 55) sont disponibles sur la base du paiement à l'utilisation. Les instances EC2 qui utilisent une activation sur site Systems Manager (ce qui n'est pas un scénario courant) sont également concernées.

A condition que votre AWS compte et votre région contiennent moins de 1 000 instances sur site dans votre environnement hybride, vous pouvez revenir au niveau des instances standard à tout moment.

Rubriques

- Activation du niveau d'instances avancées (p. 853)
- Revenir du niveau des instances avancées au niveau des instances standard (p. 858)

Activation du niveau d'instances avancées

AWS Systems Managerpropose un niveau d'instances standard et un niveau d'instances avancées pour les serveurs et VMs dans votre environnement hybride. Le niveau d'instances standard vous permet d'enregistrer 1 000 serveurs sur site au maximum ou VMs par AWS compte et par AWS région. Si vous devez enregistrer plus de 1 000 serveurs sur site ou VMs dans un seul compte et une seule région, utilisez le niveau des instances avancées. Dans un environnement hybride, vous pouvez activer autant d'instances gérées que vous le souhaitez dans le niveau d'instances avancées. Toutefois, toutes les instances configurées pour Systems Manager à l'aide du processus d'activation d'instance gérée décrit précédemment dans Créer une activation d'instance gérée pour un environnement hybride (p. 55) sont disponibles sur la base du paiement à l'utilisation. Les instances EC2 qui utilisent une activation sur site Systems Manager (ce qui n'est pas un scénario courant) sont également concernées.

Note

- Les instances avancées vous permettent également de vous connecter à vos machines hybrides en utilisant AWS Systems Manager Session Manager. Session Manager fournit l'accès au shell interactif à vos instances. Pour plus d'informations, consultez la section AWS Systems Manager Session Manager (p. 869).
- La limite des instances standard s'applique également aux instances EC2 qui utilisent une activation sur site Systems Manager (ce qui n'est pas un scénario courant).

- L'application des correctifs aux applications Microsoft est disponible uniquement sur les instances EC2 et dans le niveau d'instances avancées. Pour corriger les applications Microsoft sur les serveurs sur site et VMs , vous devez activer le niveau d'instances avancées. Pour plus d'informations, consultez la section À propos des correctifs d'applications sur Windows Server (p. 1108).

Cette section décrit comment configurer votre environnement hybride de sorte à utiliser le niveau d'instances avancées.

Avant de commencer

Passez en revue les informations de tarification pour les instances avancées. Les instances avancées sont une fonctionnalité au niveau du compte. Les instances avancées sont disponibles en fonction de l'utilisation. Pour de plus amples informations, veuillez consulter Tarification AWS Systems Manager.

Configuration des autorisations pour activer le niveau des instances avancées.

Vérifiez que vous êtes autorisé dans AWS Identity and Access Management (IAM) à basculer dans votre environnement du niveau d'instances standard au niveau d'instances avancées. La AdministratorAccess stratégie doit être attachée à votre utilisateur, groupe ou rôle IAM. Sinon, vous devez être autorisé à modifier le paramètre de service du niveau d'activation Systems Manager. Le paramètre de niveau d'activation utilise les actions d'API suivantes :

- GetServiceSetting
- UpdateServiceSetting
- ResetServiceSetting

Utilisez la procédure suivante pour ajouter une stratégie IAM en ligne à un compte utilisateur. Cette stratégie permet à un utilisateur d'afficher le paramètre actuel de niveau d'instance gérée. Cette stratégie permet également à l'utilisateur de modifier ou de réinitialiser le paramètre actuel dans le AWS compte et la région spécifiés.

1. Connectez-vous à la AWS Management Console et ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation, sélectionnez Users (Utilisateurs).
3. Dans la liste, sélectionnez le nom de l'utilisateur auquel intégrer une stratégie.
4. Choisissez l'onglet Autorisations.
5. Sur le côté droit de la page, sous Permission policies (Stratégies d'autorisation), choisissez Add inline policy (Ajouter une stratégie en ligne).
6. Choisissez l'onglet JSON.
7. Remplacez le contenu par défaut par ce qui suit :

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

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

```
            "Action": [
                "ssm:ResetServiceSetting",
                "ssm:UpdateServiceSetting"
            ],
            "Resource": "arn:aws:ssm:region:aws-account-id:servicesetting/ssm/managed-
instance/activation-tier"
        }
    ]
}
```

8. Choisissez Examiner une stratégie.
9. Sur la page Examiner une stratégie, dans le champ Nom, saisissez un nom pour la stratégie en ligne. Exemple : **`Managed-Instances-Tier`**.
10. Choisissez Create policy (Créer une stratégie).

Les administrateurs peuvent spécifier une autorisation en lecture seule en affectant la stratégie en ligne suivante au compte de l'utilisateur.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetServiceSetting"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Deny",
            "Action": [
                "ssm:ResetServiceSetting",
                "ssm:UpdateServiceSetting"
            ],
            "Resource": "*"
        }
    ]
}
```

Pour de plus amples informations sur la création et la modification de stratégies IAM, veuillez consulter Création de stratégies IAM dans le IAM Guide de l'utilisateur.

Activation du niveau d'instances avancées (console)

La procédure suivante vous montre comment utiliser la Systems Manager console pour modifier tous les serveurs sur site qui ont été ajoutés à l'aide de l'activation d'instance gérée, dans le VMs compte et la région spécifiés, afin d'utiliser le niveau d'instances avancées.AWS

> **Important**
>
> La procédure suivante décrit comment modifier un paramètre au niveau du compte. Cette modification entraîne des frais qui seront facturés à votre compte.

Pour activer le niveau d'instances avancées (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Managed instances (Instances gérées).
3. Choisissez l'onglet Settings (Paramètres).

Si vous ne voyez pas l'onglet Settings (Paramètres), procédez comme suit :

1. Vérifiez que la console est ouverte dans la AWS région où vous avez créé vos instances gérées. Vous pouvez changer de région à l'aide de la liste figurant dans la partie supérieure droite de la console.
2. Vérifiez que vos instances respectent les exigences de Systems Manager. Pour plus d'informations, consultez la section Prérequis d'Systems Manager (p. 15).
3. Pour les serveurs et VMs dans un environnement hybride, vérifiez que vous avez terminé le processus d'activation. Pour plus d'informations, consultez la section Configuration d'AWS Systems Manager pour les environnements hybrides (p. 49).

4. Choisissez Modifier les paramètres de compte.
5. Vérifiez les informations contenues dans la fenêtre sur la modification des paramètres de compte puis, si vous approuvez, choisissez l'option pour accepter et continuer.

Le processus consistant à déplacer toutes les instances du niveau d'instances standard au niveau d'instances avancées peut prendre plusieurs minutes.

> **Note**
>
> Pour de plus amples informations sur le retour au niveau des instances standard, veuillez consulter Revenir du niveau des instances avancées au niveau des instances standard (p. 858).

Activation du niveau d'instances avancées (AWS CLI)

La procédure suivante vous montre comment utiliser l' AWS CLI pour modifier tous les serveurs sur site qui ont été ajoutés à l'aide de l'activation d'instance gérée, dans le VMs compte et la région spécifiés, afin d'utiliser le niveau d'instances avancées.AWS

> **Important**
>
> La procédure suivante décrit comment modifier un paramètre au niveau du compte. Cette modification entraîne des frais qui seront facturés à votre compte.

Pour activer le niveau d'instances avancées à l'aide d'AWS CLI

1. Ouvrez la AWS CLI et exécutez la commande suivante.

   Linux

   ```
   aws ssm update-service-setting \
       --setting-id arn:aws:ssm:region:aws-account-id:servicesetting/ssm/managed-
   instance/activation-tier \
       --setting-value advanced
   ```

   Windows

   ```
   aws ssm update-service-setting ^
       --setting-id arn:aws:ssm:region:aws-account-id:servicesetting/ssm/managed-
   instance/activation-tier ^
       --setting-value advanced
   ```

   Il n'y a pas de sortie si la commande réussit.
2. Exécutez la commande suivante pour afficher les paramètres de service actuels pour les instances gérées dans le AWS compte et la région actuels.

Linux

```
aws ssm get-service-setting \
    --setting-id arn:aws:ssm:region:aws-account-id:servicesetting/ssm/managed-
instance/activation-tier
```

Windows

```
aws ssm get-service-setting ^
    --setting-id arn:aws:ssm:region:aws-account-id:servicesetting/ssm/managed-
instance/activation-tier
```

La commande renvoie des informations telles que les suivantes.

```
{
    "ServiceSetting": {
        "SettingId": "/ssm/managed-instance/activation-tier",
        "SettingValue": "advanced",
        "LastModifiedDate": 1555603376.138,
        "LastModifiedUser": "arn:aws:sts::123456789012:assumed-role/
Administrator/User_1",
        "ARN": "arn:aws:ssm:us-east-2:123456789012:servicesetting/ssm/managed-instance/
activation-tier",
        "Status": "PendingUpdate"
    }
}
```

Activation du niveau d'instances avancées (PowerShell)

La procédure suivante vous montre comment utiliser l' Outils AWS pour Windows PowerShell pour modifier tous les serveurs sur site qui ont été ajoutés à l'aide de l'activation d'instance gérée, dans le VMs compte et la région spécifiés, afin d'utiliser le niveau d'instances avancées.AWS

> **Important**
>
> La procédure suivante décrit comment modifier un paramètre au niveau du compte. Cette modification entraîne des frais qui seront facturés à votre compte.

Pour activer le niveau d'instances avancées à l'aide dePowerShell

1. Ouvrez Outils AWS pour Windows PowerShell et exécutez la commande suivante.

```
Update-SSMServiceSetting `
    -SettingId "arn:aws:ssm:region:aws-account-id:servicesetting/ssm/managed-instance/
activation-tier" `
    -SettingValue "advanced"
```

   Il n'y a pas de sortie si la commande réussit.

2. Exécutez la commande suivante pour afficher les paramètres de service actuels pour les instances gérées dans le AWS compte et la région actuels.

```
Get-SSMServiceSetting `
    -SettingId "arn:aws:ssm:region:aws-account-id:servicesetting/ssm/managed-instance/
activation-tier"
```

La commande renvoie des informations telles que les suivantes.

```
ARN:arn:aws:ssm:us-east-2:123456789012:servicesetting/ssm/managed-instance/activation-
tier
LastModifiedDate : 4/18/2019 4:02:56 PM
LastModifiedUser : arn:aws:sts::123456789012:assumed-role/Administrator/User_1
SettingId        : /ssm/managed-instance/activation-tier
SettingValue     : advanced
Status           : PendingUpdate
```

Le processus consistant à déplacer toutes les instances du niveau d'instances standard au niveau d'instances avancées peut prendre plusieurs minutes.

Note

Pour de plus amples informations sur le retour au niveau des instances standard, veuillez consulter Revenir du niveau des instances avancées au niveau des instances standard (p. 858).

Revenir du niveau des instances avancées au niveau des instances standard

Cette section décrit comment faire passer les instances hybrides actuellement exécutées du niveau des instances avancées au niveau des instances standard. Cette configuration s'applique à toutes les instances hybrides d'un AWS compte et d'une seule AWS région.

Avant de commencer

Consultez les détails importants suivants.

Note

- Vous ne pouvez pas revenir au niveau des instances standard si vous exécutez plus de 1 000 instances hybrides dans le compte et la région. Vous devez d'abord annuler l'enregistrement des instances hybrides jusqu'à ce que vous en ayez 1 000 ou moins. Les instances EC2 qui utilisent une activation sur site Systems Manager (ce qui n'est pas un scénario courant) sont également concernées. Pour plus d'informations, consultez la section Annuler l'enregistrement des instances gérées dans un environnement hybride (p. 864).
- Une fois que vous serez revenu en arrière, vous ne pourrez plus utiliser Session Manager pour accéder de manière interactive à vos instances hybrides.
- Une fois que vous serez revenu en arrière, vous ne pourrez plus utiliser Gestionnaire de correctifs pour corriger des applications Microsoft sur des serveurs hybrides et des machines virtuelles (VM).
- Le processus de retour de toutes les instances hybrides au niveau des instances standard peut prendre 30 minutes ou plus.

Cette section décrit comment rétablir toutes les instances hybrides d'un AWS compte et d'une région du niveau des instances avancées au niveau des instances standard.

Revenir au niveau des instances standard (console)

La procédure suivante vous montre comment utiliser la Systems Manager console pour modifier tous les serveurs sur site et VMs dans votre environnement hybride afin d'utiliser le niveau des instances standard dans le AWS compte et la région spécifiés.

Pour revenir au niveau des instances standard (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Managed instances (Instances gérées).
3. Choisissez l'onglet Settings (Paramètres).
4. Choisissez Modifier les paramètres de compte.
5. Vérifiez les informations contenues dans la fenêtre sur la modification des paramètres de compte puis, si vous approuvez, choisissez l'option pour accepter et continuer.

Revenir au niveau des instances standard (AWS CLI)

La procédure suivante vous montre comment utiliser l' AWS CLI pour modifier tous les serveurs sur site et VMs dans votre environnement hybride afin d'utiliser le niveau des instances standard dans le AWS compte et la région spécifiés.

Pour revenir au niveau des instances standard à l'aide de AWS CLI

1. Ouvrez la AWS CLI et exécutez la commande suivante.

   Linux

   ```
   aws ssm update-service-setting \
       --setting-id arn:aws:ssm:region:aws-account-id:servicesetting/ssm/managed-
   instance/activation-tier \
       --setting-value standard
   ```

   Windows

   ```
   aws ssm update-service-setting ^
       --setting-id arn:aws:ssm:region:aws-account-id:servicesetting/ssm/managed-
   instance/activation-tier ^
       --setting-value standard
   ```

   Il n'y a pas de sortie si la commande réussit.
2. Exécutez la commande suivante 30 minutes plus tard pour afficher les paramètres des instances gérées dans le AWS compte et la région actuels.

   Linux

   ```
   aws ssm get-service-setting \
       --setting-id arn:aws:ssm:region:aws-account-id:servicesetting/ssm/managed-
   instance/activation-tier
   ```

   Linux

   ```
   aws ssm get-service-setting ^
       --setting-id arn:aws:ssm:region:aws-account-id:servicesetting/ssm/managed-
   instance/activation-tier
   ```

   La commande renvoie des informations telles que les suivantes.

   ```
   {
   ```

```
    "ServiceSetting": {
        "SettingId": "/ssm/managed-instance/activation-tier",
        "SettingValue": "standard",
        "LastModifiedDate": 1555603376.138,
        "LastModifiedUser": "System",
        "ARN": "arn:aws:ssm:us-east-2:123456789012:servicesetting/ssm/managed-instance/
activation-tier",
        "Status": "Default"
    }
}
```

Le statut devient Par défaut une fois la demande approuvée.

Revenir au niveau des instances standard (PowerShell)

La procédure suivante vous montre comment utiliser Outils AWS pour Windows PowerShell pour modifier tous les serveurs sur site et VMs dans votre environnement hybride afin d'utiliser le niveau des instances standard dans le AWS compte et la région spécifiés.

Pour revenir au niveau des instances standard à l'aide dePowerShell

1. Ouvrez Outils AWS pour Windows PowerShell et exécutez la commande suivante.

```
Update-SSMServiceSetting `
    -SettingId "arn:aws:ssm:region:aws-account-id:servicesetting/ssm/managed-instance/
activation-tier" `
    -SettingValue "standard"
```

   Il n'y a pas de sortie si la commande réussit.

2. Exécutez la commande suivante 30 minutes plus tard pour afficher les paramètres des instances gérées dans le AWS compte et la région actuels.

```
Get-SSMServiceSetting `
    -SettingId "arn:aws:ssm:region:aws-account-id:servicesetting/ssm/managed-instance/
activation-tier"
```

   La commande renvoie des informations telles que les suivantes.

```
ARN: arn:aws:ssm:us-east-2:123456789012:servicesetting/ssm/managed-instance/activation-
tier
LastModifiedDate : 4/18/2019 4:02:56 PM
LastModifiedUser : System
SettingId        : /ssm/managed-instance/activation-tier
SettingValue     : standard
Status           : Default
```

   Le statut devient Par défaut une fois la demande approuvée.

Réinitialisation de mots de passe sur des instances gérées

Vous pouvez réinitialiser le mot de passe pour n'importe quel utilisateur sur une instance gérée. Cela inclut les instances Amazon EC2, les serveurs sur site et les machines virtuelles qui sont gérés par AWS Systems Manager. La fonctionnalité de réinitialisation de mot de passe est basée sur la fonctionnalité

Session Manager AWS Systems Manager. Vous pouvez utiliser cette fonctionnalité pour vous connecter à des instances sans ouvrir des ports entrants, maintenir des hôtes bastion ou gérer des clés SSH.

Ainsi, cette option de réinitialisation de mot de passe est utile lorsqu'un utilisateur a oublié un mot de passe, ou lorsque vous souhaitez rapidement mettre à jour un mot de passe sans établir une connexion SSH ou RDP vers l'instance.

Prerequisites

Pour que vous puissiez réinitialiser le mot de passe sur une instance, les exigences suivantes doivent être satisfaites :

- L'instance sur laquelle vous voulez modifier un mot de passe doit être une instance gérée Systems Manager. Cela signifie que Agent SSM est installé sur l'instance. (Agent SSMLa version 2.3 668.0 ou ultérieure est requise pour modifier les mots de passe.) Pour plus d'informations sur l'installation ou la mise à jour Agent SSM de , consultez Utilisation des Agent SSM (p. 72) .
- La fonctionnalité de réinitialisation de mot de passe utilise la AWS Session Manager configuration configurée pour que votre compte se connecte à l'instance. Par conséquent, les prérequis pour l'utilisation de Session Manager doivent avoir été satisfaits pour votre compte dans la région actuelle. Pour plus d'informations, consultez la section Configuration d'Session Manager (p. 873).

  Note

  La prise en charge de Session Manager pour les serveurs sur site est assurée pour le niveau des instances avancées uniquement. Pour plus d'informations, consultez la section Activation du niveau d'instances avancées (p. 853).
- L'utilisateur AWS qui modifie le mot de passe doit disposer de l'autorisation `ssm:SendCommand` pour l'instance. Pour plus d'informations, consultez la section Restriction de l'accès de la fonctionnalité Fonctionnalité Exécuter la commande en fonction des balises d'instance (p. 943).

Restriction de l'accès

Vous pouvez limiter la capacité d'un utilisateur à réinitialiser des mots de passe pour des instances spécifiques. Pour ce faire, utilisez des stratégies basées sur une identité pour l'action Session Manager `ssm:StartSession` avec le document SSM `AWS-PasswordReset`. Pour plus d'informations, consultez la section Contrôler les accès de session utilisateur aux instances (p. 882).

Chiffrement de données

Vous devez activer le chiffrement de bout en bout AWS Key Management Service (AWS KMS) pour que les données Session Manager utilisent l'option de réinitialisation de mot de passe pour les instances gérées. Pour plus d'informations, consultez la section Activer le chiffrement de clé AWS KMS des données de session (console) (p. 902).

Réinitialiser un mot de passe sur une Instance gérée

Vous pouvez réinitialiser un mot de passe sur une instance gérée Systems Manager à l'aide de la console Managed Instances (Instances gérées) AWS Systems Manager ou de l'AWS CLI.

Pour modifier le mot de passe sur une instance gérée (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Instances gérées.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Instances gérées.

3. Choisissez le bouton en regard de l'instance qui a besoin d'un nouveau mot de passe.
4. Dans le menu Actions, choisissez Reset password (Réinitialiser un mot de passe).
5. Pour User name (Nom d'utilisateur), tapez le nom de l'utilisateur pour lequel vous modifiez le mot de passe. Il peut s'agit de tout nom d'utilisateur qui dispose d'un compte sur l'instance.
6. Choisissez Submit.
7. Suivez les instructions dans la fenêtre de commande Enter new password (Entrer un nouveau mot) pour spécifier le nouveau mot de passe.

   Note

   Si la version de Agent SSM sur l'instance ne prend pas en charge les réinitialisations de mot de passe, vous êtes invité à installer une version prise en charge avec Fonctionnalité Exécuter la commande.

Pour réinitialiser le mot de passe sur une instance gérée (interface de ligne de commande)

1. Pour réinitialiser le mot de passe d'un utilisateur sur une instance gérée, exécutez la commande suivante.

   Note

   Pour que vous puissiez utiliser l'AWS CLI pour réinitialiser un mot de passe, Session Manager plugin doit être installé sur votre ordinateur local. Pour plus d'informations, consultez la section (Facultatif) Installer le Session Manager plugin pour l'AWS CLI (p. 917).

   Linux

   ```
   aws ssm start-session \
       --target instance-id \
       --document-name "AWS-PasswordReset" \
       --parameters "{"username": "user-name"}"
   ```

   Windows

   ```
   aws ssm start-session ^
       --target instance-id ^
       --document-name "AWS-PasswordReset" ^
       --parameters "{"username": "user-name"}"
   ```

   instance-id représente l'ID d'une instance configurée pour être utilisée avec Systems Manager et sa Session Manager capacité.

   user-name représente le nom de l'utilisateur pour lequel vous souhaitez réinitialiser le mot de passe sur l'instance.
2. Suivez les instructions dans la fenêtre de commande Enter new password (Entrer un nouveau mot) pour spécifier le nouveau mot de passe.

Résoudre les problèmes de réinitialisation de mot de passe sur des instances gérées

De nombreux problèmes de réinitialisation de mot de passe peuvent être résolus en vous assurant que vous avez rempli les Prérequis pour la réinitialisation de mot de passe (p. 861). Pour les autres problèmes, utilisez les informations suivantes pour vous aider à résoudre les problèmes de réinitialisation de mot de passe.

Rubriques

- Instance non disponible (p. 863)
- Agent SSM n'est pas à jour (console) (p. 863)
- Les options de réinitialisation de mot de passe n'apparaissent pas (interface de ligne de commande) (p. 863)
- Pas d'autorisation pour exécuter ssm:SendCommand (p. 864)
- Message d'erreur Session Manager (p. 864)

Instance non disponible

Problème : Vous souhaitez réinitialiser le mot de passe d'une instance EC2 sur la page de la console Instances gérées, mais l'instance ne figure pas dans la liste.

- Solution: L'instance à laquelle vous souhaitez vous connecter n'est peut-être pas configurée pour être utilisée avec le AWS Systems Manager service. Pour utiliser une instance EC2 avec Systems Manager, un profil d'instance IAM accordant l'autorisation Systems Manager d'effectuer des actions sur vos instances doit être attaché à l'instance. Pour plus d'informations, consultez la section Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). Pour utiliser un serveur sur site ou une machine virtuelle que vous avez activé pour une utilisation avec Systems Manager, vous devez créer un rôle de service IAM qui accorde à Systems Manager l'autorisation d'effectuer des actions sur vos machines. Pour plus d'informations, consultez Créer un rôle de service IAM pour un environnement hybride (p. 50) . La Session Manager prise en charge d' (pour les serveurs sur site VMs et est fournie pour le niveau des instances avancées uniquement. Pour plus d'informations, consultez Activation du niveau d'instances avancées (p. 853) .)

Agent SSM n'est pas à jour (console)

Problème : Un message indique que la version d' Agent SSM ne prend pas en charge la fonctionnalité de réinitialisation de mot de passe.

- Solution: La version 2.3.668.0 ou ultérieure de Agent SSM est requise pour effectuer des réinitialisations de mot de passe. Dans la console, vous pouvez lancer le processus de mise à jour de l'agent sur l'instance en choisissant Mettre à jour l'agent SSM.

  Une version mise à jour de l'Agent SSM est publiée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Si une version antérieure de l'agent est en cours d'exécution sur une instance, certains processus de l'Agent SSM peuvent échouer. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir l'Agent SSM à jour sur vos instances. Pour de plus amples informations, veuillez consulter Automatisation des mises à jour versAgent SSM (p. 109). Pour être alerté des mises à jour de Agent SSM, abonnez-vous à la page Agent SSM Release Notes sur GitHub.

Les options de réinitialisation de mot de passe n'apparaissent pas (interface de ligne de commande)

Problème : Vous vous connectez avec succès à une instance à l'aide de la AWS CLI `start-session` commande . Vous avez spécifié le document SSM `AWS-PasswordReset` et vous avez fourni un nom utilisateur valide, mais les invites pour modifier le mot de passe n'apparaissent pas.

- Solution: La version d' Agent SSM sur l'instance n'est pas à jour. La version 2.3.668.0 ou suivante est requise pour effectuer des réinitialisations de mot de passe.

  Une version mise à jour de l'Agent SSM est publiée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes.

Si une version antérieure de l'agent est en cours d'exécution sur une instance, certains processus de l'Agent SSM peuvent échouer. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir l'Agent SSM à jour sur vos instances. Pour de plus amples informations, veuillez consulter Automatisation des mises à jour versAgent SSM (p. 109). Pour être alerté des mises à jour de Agent SSM, abonnez-vous à la page Agent SSM Release Notes sur GitHub.

Pas d'autorisation pour exécuter `ssm:SendCommand`

Problème : Vous essayez de vous connecter à une instance pour modifier son mot de passe, mais vous recevez un message d'erreur indiquant que vous n'êtes pas autorisé à exécuter `ssm:SendCommand` sur l'instance.

- Solution: Votre stratégie d'utilisateur IAM doit inclure l'autorisation d'exécuter la `ssm:SendCommand` commande. Pour plus d'informations, consultez la section Restriction de l'accès de la fonctionnalité Fonctionnalité Exécuter la commande en fonction des balises d'instance (p. 943).

Message d'erreur Session Manager

Problème : Vous recevez un message d'erreur lié à Session Manager .

- Solution: La prise en charge de la réinitialisation du mot de passe nécessite que Session Manager soit correctement configuré. Pour plus d'informations, consultez Configuration d'Session Manager (p. 873) et Dépannage de Session Manager (p. 938).

Annuler l'enregistrement des instances gérées dans un environnement hybride

Si vous ne souhaitez plus gérer un serveur local ou une machine virtuelle (VM) à l'aide de AWS Systems Manager, vous pouvez annuler son enregistrement. L'annulation de l'enregistrement d'une machine hybride la supprime de la liste des instances gérées dans Systems Manager. Agent SSM exécuté sur la machine hybride ne pourra pas actualiser son jeton d'autorisation car il n'est plus enregistré. Agent SSM hibernera et réduira sa fréquence ping à Systems Manager dans le cloud à une fois par heure.

Vous pouvez réenregistrer un serveur local ou une machine virtuelle à tout moment. Systems Manager stocke l'historique des commandes d'une instance gérée dont l'enregistrement a été annulé pendant 30 jours.

La procédure suivante décrit comment annuler l'enregistrement d'une machine hybride à l'aide de la AWS Systems Manager console. Pour de plus amples informations sur la façon de procéder à l'aide de la AWS CLI, veuillez consulter deregister-managed-instance.

Pour annuler l'enregistrement d'une machine hybride (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Managed instances (Instances gérées).
3. Dans la liste Instances gérées choisissez l'instance dont vous souhaitez annuler l'enregistrement, puis choisissez Actions, Annuler l'enregistrement de cette instance gérée.
4. Passez en revue les informations contenues dans la fenêtre contextuelle Annuler l'enregistrement de cette instance gérée puis, si vous approuvez, choisissez Annuler l'enregistrement.

Dépannage de la disponibilité Amazon EC2 des instances gérées

Pour plusieurs Systems Manager opérations, vous pouvez choisir de sélectionner manuellement les instances sur lesquelles vous allez exécuter l'opération. Les exemples incluent l'exécution d'une Fonctionnalité Exécuter la commande commande, la spécification de cibles de fenêtre de maintenance, l'installation de Distributor packages et la connexion à l'instance à l'aide d' Session Manager . Une fois que vous avez spécifié que vous souhaitez choisir les instances manuellement, une liste des instances gérées sur lesquelles vous pouvez choisir d'exécuter l'opération s'affiche.

Cette rubrique fournit des informations pour vous aider à diagnostiquer pourquoi une instance Amazon Elastic Compute Cloud (Amazon EC2) que vous avez confirmée est en cours d'exécution n'apparaît pas dans vos listes d'instances gérées dans Systems Manager .

Pour qu'une instance EC2 soit gérée par Systems Manager et disponible dans des listes d'instances gérées, elle doit répondre à trois exigences principales :

1. Agent SSMdoit être installé et en cours d'exécution sur une instance avec un système d'exploitation pris en charge.

   Note

   Certaines Amazon Machine Images (AMI) sont configurées pour lancer des instances avec Agent SSM (p. 72) préinstallé. (Vous pouvez également configurer une AMI personnalisée pour préinstaller Agent SSM .)
   L'Agent SSM est préinstallé par défaut sur toutes les images machine Amazon (AMI) suivantes :

   - Amazon Linux
   - Amazon Linux 2
   - AMIs optimisées pour ECS Amazon Linux 2
   - Ubuntu Server 16.04, 18.04 et 20.04
   - AMIs Windows Server 2008-2012 R2 publiées à partir de novembre 2016
   - Windows Server 2016 et 2019

     Note

     L'Agent SSM n'est pas installé sur toutes les AMIs basées sur Amazon Linux ou Amazon Linux 2. Par exemple, l'Agent SSM n'est pas préinstallé sur les AMIs optimisées pour EKS basées sur Amazon Linux 2.
2. Un profil d'instance AWS Identity and Access Management (IAM) qui fournit les autorisations requises pour que l'instance puisse communiquer avec le Systems Manager service doit être attaché à l'instance.
3. Agent SSMdoit être en mesure de se connecter à un Systems Manager point de terminaison pour s'inscrire lui-même auprès du service. Par la suite, l'instance doit être disponible pour le service, ce qui est confirmé par l'envoi d'un signal toutes les cinq minutes pour vérifier l'état de l'instance.

Une fois que vous avez vérifié qu'une instance EC2 est en cours d'exécution, vous pouvez utiliser la commande suivante pour vérifier si Agent SSM sur l'un de ces types d'instance s'est bien enregistrée auprès du Systems Manager service. Cette commande ne renvoie pas de résultats tant qu'une inscription n'a pas réussi.

Linux

```
aws ssm describe-instance-associations-status \
    --instance-id instance-id
```

Windows

```
aws ssm describe-instance-associations-status ^
    --instance-id instance-id
```

PowerShell

```
Get-SSMInstanceAssociationsStatus `
    -InstanceId instance-id
```

Si l'enregistrement a réussi et que l'instance est désormais disponible pour Systems Manager les opérations , la commande renvoie des résultats similaires à ce qui suit.

```
{
    "InstanceAssociationStatusInfos": [
        {
            "AssociationId": "fa262de1-6150-4a90-8f53-d7eb5EXAMPLE",
            "Name": "AWS-GatherSoftwareInventory",
            "DocumentVersion": "1",
            "AssociationVersion": "1",
            "InstanceId": "i-02573cafcfEXAMPLE",
            "Status": "Pending",
            "DetailedStatus": "Associated"
        },
        {
            "AssociationId": "f9ec7a0f-6104-4273-8975-82e34EXAMPLE",
            "Name": "AWS-RunPatchBaseline",
            "DocumentVersion": "1",
            "AssociationVersion": "1",
            "InstanceId": "i-02573cafcfEXAMPLE",
            "Status": "Queued",
            "AssociationName": "SystemAssociationForScanningPatches"
        }
    ]
}
```

Si l'enregistrement n'est pas encore terminé ou a échoué, la commande renvoie des résultats similaires à ce qui suit :

```
{
    "InstanceAssociationStatusInfos": []
}
```

Si la commande ne renvoie pas de résultats après cinq minutes environ, utilisez les informations suivantes pour vous aider à résoudre les problèmes liés à vos instances gérées.

Rubriques

- Solution 1 : Vérifiez que Agent SSM est installé et en cours d'exécution sur l'instance (p. 867)
- Solution 2 : Vérifier qu'un rôle dIAM'instance a été spécifié pour l'instance (p. 867)
- Solution 3 : Vérification de la connectivité du point de terminaison de service (p. 868)
- Solution 4 : Vérification de la prise en charge du système d'exploitation cible (p. 868)
- Solution 5 : Vérifiez que vous travaillez dans la même région AWS que l'instance EC2 (p. 868)

Solution 1 : Vérifiez que Agent SSM est installé et en cours d'exécution sur l'instance

Assurez-vous que la dernière version de Agent SSM est installée et en cours d'exécution sur l'instance.

Pour vérifier si Agent SSM est installé et en cours d'exécution sur une instance EC2, consultez Vérification de Agent SSM l'état et démarrage de l'agent (p. 104) .

Pour installer ou réinstaller Agent SSM sur une instance EC2, consultez les rubriques suivantes :

- Installation et configuration de l'Agent SSM sur les instances EC2 pour Linux (p. 75)
- Installation et configuration Agent SSM de sur les instances EC2 pourWindows Server (p. 100)

Solution 2 : Vérifier qu'un rôle dIAM'instance a été spécifié pour l'instance

Vérifiez que l'instance est configurée avec un rôle AWS Identity and Access Management (IAM) qui permet à l'instance de communiquer avec l'Systems ManagerAPI . Vérifiez également que votre compte utilisateur est associé à une stratégie d'approbation utilisateur IAM qui permet à votre compte afin de communiquer avec l'API Systems Manager.

Pour vérifier si un profil d'instance avec les autorisations nécessaires est attaché à l'instance

1. Ouvrez la console Amazon EC2 à l'adresse https://console.aws.amazon.com/ec2/.
2. Dans le panneau de navigation, choisissez Instances.
3. Sélectionnez l'instance pour laquelle rechercher un profil d'instance.
4. Sous l'onglet Description dans le volet du bas, recherchez le rôle IAM et choisissez le nom du rôle.
5. Sur la page récapitulative du rôle pour le profil d'instance, dans l'onglet Autorisations, assurez-vous que `AmazonSSMManagedInstanceCore` est répertorié sous Stratégies d'autorisations.

   Si une stratégie personnalisée est utilisée à la place, assurez-vous qu'elle fournit les mêmes autorisations qu' `AmazonSSMManagedInstanceCore` .

   Permet d`AmazonSSMManagedInstanceCore`'ouvrir dans la console

   Pour plus d'informations sur les autres stratégies qui peuvent être attachées à un profil d'instance pour Systems Manager , consultez Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34) .

Pour créer et attacher un profil d'instance avec les autorisations nécessaires à une nouvelle instance EC2

- Effectuez les tâches décrites dans les rubriques suivantes :
  - Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34)
  - Lancer une instance qui utilise le profil d'instance Systems Manager (console) (p.)

Pour attacher un profil d'instance existant avec les autorisations nécessaires à une instance EC2 existante

- Exécutez la tâche décrite dans la rubrique suivante :
  - Attacher le profil d'instance Systems Manager à une instance existante (console) (p.)

Solution 3 : Vérification de la connectivité du point de terminaison de service

Vérifiez que l'instance est connectée aux points de terminaison de Systems Manager service. Cette connectivité est fournie en créant et en configurant des points de terminaison de VPC pour Systems Manager, ou en autorisant le trafic sortant HTTPS (port 443) vers les points de terminaison de service.

Dans de nombreux cas, lorsque vous créez une instance EC2, le point de terminaison de Systems Manager service de la région AWS dans laquelle l'instance est utilisée pour enregistrer l'instance. Cependant, si vous utilisez un Virtual Private Cloud (VPC) et que l'instance EC2 a été créée dans un sous-réseau privé, les points de terminaison de service ne sont pas fournis automatiquement. Vous devez configurer des points de terminaison d'interface pour votre VPC à la place.

Pour plus d'informations, consultez la section (Facultatif) Créer un point de terminaison de VPC (p. 42).

Solution 4 : Vérification de la prise en charge du système d'exploitation cible

Vérifiez que l'opération que vous avez choisie peut être exécutée sur le type d'instance que vous vous attendez à voir répertorié. Certaines Systems Manager opérations peuvent cibler uniquement des instances Windows ou des instances Linux. Par exemple, les documents Systems Manager (SSM) `AWS-InstallPowerShellModule` peuvent être exécuté`AWS-ConfigureCloudWatch`s uniquement sur des instances Windows. Dans la page Exécuter une commande, si vous choisissez l'un de ces documents et sélectionnez Choisir les instances manuellement, seules vos instances Windows sont répertoriées et disponibles pour la sélection.

Solution 5 : Vérifiez que vous travaillez dans la même région AWS que l'instance EC2

Les instances EC2 sont créées et disponibles dans des régions AWS spécifiques, comme Région USA Est (Ohio) (us-east-2) ou Région Europe (Irlande) (eu-west-1). Veillez à travailler dans la même région AWS que l'instance EC2 que vous souhaitez utiliser. Pour plus d'informations, consultez Choix d'une région dans Mise en route avec AWS Management Console .

Activations hybrides AWS Systems Manager

Pour configurer les serveurs et les machines virtuelles (VM) de votre environnement hybride en tant qu'instances gérées, vous créez une activation hybride d'instance gérée. Une fois que vous avez terminé l'activation, vous recevez un identifiant et un code d'activation. Cette combinaison code/ID fonctionne comme un ID d'accès et une clé secrète Amazon EC2. Elle sécurise l'accès au service Systems Manager à partir de vos instances gérées.

Pour plus d'informations sur la configuration de serveurs sur site et de VM en tant qu'instances gérées, consultez Configuration d'AWS Systems Manager pour les environnements hybrides (p. 49).

À propos de AWS Systems Manager Niveaux d’instances

AWS Systems Manager offre un niveau d'instances standard et un niveau d'instances avancées pour les serveurs et machines virtuelles de votre environnement hybride. Le niveau des instances standard vous permet d'enregistrer 1 000 serveurs ou machines virtuelles au maximum par compte AWS et par région AWS. Si vous avez besoin d'enregistrer plus de 1 000 serveurs ou machines virtuelles dans un seul compte et une seule région, utilisez le niveau des instances avancées. Vous pouvez créer autant d'instances que vous le souhaitez dans le niveau des instances avancées, mais toutes les instances configurées pour

Systems Manager sont facturées en fonction de l'utilisation. Pour plus d'informations sur l'activation des instances avancées, consultez Activation du niveau d'instances avancées (p. 853). Pour de plus amples informations sur la tarification, veuillez consulter Tarification d'AWS Systems Manager.

Note

- Les instances avancées vous permettent également de vous connecter à vos machines hybrides en utilisant AWS Systems Manager Session Manager. Session Manager fournit l'accès au shell interactif à vos instances. Pour plus d'informations, consultez AWS Systems Manager Session Manager (p. 869).
- Le quota d'instances standard s'applique également aux instances EC2 qui utilisent une activation sur site de Systems Manager (ce qui n'est pas un scénario courant).
- L'application des correctifs aux applications Microsoft est disponible uniquement sur les instances EC2 et dans le niveau d'instances avancées. Pour corriger les applications Microsoft sur les serveurs et les machines virtuelles sur site, vous devez activer le niveau d'instances avancées. Pour plus d'informations, consultez À propos des correctifs d'applications sur Windows Server (p. 1108).

AWS Systems Manager Session Manager

Session Manager est un de AWS Systems Manager qui vous permet de gérer vos Amazon Elastic Compute Cloud (Amazon EC2les instances , instances sur site, et machines virtuelles via un shell interactif basé sur un navigateur et en un clic ou via AWS Command Line Interface (AWS CLI. Session Manager permet une gestion d'instances sécurisée et pouvant faire l'objet d'un audit sans avoir besoin d'ouvrir des ports entrants maintenir les hôtes bastion, ou gérer les clés SSH. Session Manager facilite également la conformité aux stratégies d'entreprise qui nécessitent un accès contrôlé aux instances. les pratiques de sécurité strictes et des journaux pouvant faire l'objet d'un audit complet avec les détails d'accès à l'instance, tout en fournissant aux utilisateurs finaux un accès multiplateforme en un clic simple à vos instances gérées.

Comment mon organisation peut-elle tirer parti de Session Manager ?

Session Manager offre les avantages suivants :

- Contrôle centralisé des accès aux instances à l'aide IAM de stratégies

  Les administrateurs disposent d'un point central pour accorder et révoquer les accès aux instances. En utilisant uniquement les stratégies AWS Identity and Access Management (IAM), vous pouvez contrôler quels utilisateurs ou groupes au sein de votre organisation peuvent utiliser Session Manager et à quelles instances ils peuvent accéder.
- Aucun port entrant ouvert et aucune nécessité d'ouvrir des hôtes bastion ou des clés SSH

  Le fait de laisser des ports SSH entrants et PowerShell des ports distants ouverts sur vos instances augmente considérablement le risque que des entités exécutent des commandes non autorisées ou malveillantes sur les instances. vous Session Manager aide à améliorer votre niveau de sécurité en vous permettant de fermer ces ports entrants, ce qui vous libère de la gestion des clés et des certificats SSH, des hôtes bastion et des zones de saut.
- Accès en un clic aux instances depuis la console et l'interface de ligne de commande

  À l'aide de la console AWS Systems Manager ou de la console Amazon EC2, vous pouvez démarrer une session d'un simple clic. A l'aide de l'AWS CLI, vous pouvez également démarrer une session qui exécute une seule commande ou une séquence de commandes. Étant donné que les autorisations pour

les instances sont fournies via IAM des stratégies au lieu de clés SSH ou d'autres mécanismes, le temps de connexion est considérablement réduit.

- Réacheminement de port

  Redirigez n'importe quel port de votre instance distante vers un port local sur un client. Après cela, connectez-vous au port local et accédez à l'application serveur qui s'exécute sur l'instance.

- Prise en charge multiplateforme de Windows et Linux

  Session Manager offre une prise en charge de Windows et Linux, à partir d'un simple outil. Par exemple, vous n'avez pas besoin d'utiliser un client SSH pour les instances Linux, ni une connexion RDP pour les instances Windows Server.

- Activité de session de journalisation et d'audit

  Pour satisfaire aux exigences opérationnelles ou de sécurité de votre organisation, vous pouvez avoir besoin de fournir un enregistrement des différentes connexions à vos instances et des commandes qui y ont été exécutées. Vous pouvez également recevoir des notifications lorsqu'un utilisateur de votre organisation démarre ou termine une activité de session.

  Des fonctionnalités de journalisation et d'audit sont fournies par le biais de l'intégration aux AWS services suivants :

  - AWS CloudTrail - AWS CloudTrail capture des informations sur les appels d'API Session Manager effectués dans votre compte AWS et les écrit dans des fichiers journaux stockés dans un compartiment S3 que vous spécifiez. Un seul compartiment est utilisé pour tous les journaux CloudTrail de votre compte. Pour plus d'informations, consultez la section Journalisation des appels d'API AWS Systems Manager avec AWS CloudTrail (p. 1336).
  - Amazon Simple Storage Service – Vous pouvez choisir de stocker les données des journaux de session dans un compartiment S3 de votre choix à des fins d'audit. Les données des journaux peuvent être envoyées à votre compartiment S3 avec ou sans chiffrement à l'aide de votre clé AWS Key Management Service (AWS KMS). Pour plus d'informations, consultez la section Journalisation des données de session avec Amazon S3 (console) (p. 931).
  - Amazon CloudWatch Logs – CloudWatch Logs vous permet de surveiller et de stocker les fichiers journaux, et d'y accéder, à partir de différents services AWS. Vous pouvez envoyer des données de journaux de session à un groupe de journaux CloudWatch Logs à des fins d'audit. Les données de journaux peuvent être envoyées à votre groupe de journaux avec ou sans chiffrement AWS KMS à l'aide de votre clé AWS KMS. Pour plus d'informations, consultez la section Journalisation des données de session avec Amazon CloudWatch Logs (console) (p. 932).
  - Amazon EventBridge et Amazon Simple Notification Service – EventBridge vous permet de détecter à quel moment les changements ont lieu pour les ressources AWS que vous spécifiez. Vous pouvez créer une règle pour détecter le démarrage ou l'arrêt d'une session par un utilisateur de votre organisation, puis recevoir une notification via Amazon SNS (par exemple, un SMS ou un e-mail) à propos de l'événement. Vous pouvez également configurer un CloudWatch événement pour initier d'autres réponses. Pour plus d'informations, consultez la section Surveillance de l'activité de la session avec Amazon EventBridge (console) (p. 933).

    Note

    La journalisation et l'audit ne sont pas disponibles pour les sessions Session Manager qui se connectent via le transfert de port ou SSH. Cela est dû au fait que SSH crypte toutes les données de session et que Session Manager sert uniquement de tunnel pour les connexions SSH.

À qui est destiné Session Manager ?

- À tout client AWS qui souhaite améliorer son niveau de sécurité et d'audit, réduire ses coûts opérationnels en centralisant le contrôle d'accès aux instances et réduire l'accès aux instances entrantes.

- Les experts en sécurité de l'information qui souhaitent surveiller et suivre l'accès aux instances et leur activité, fermer les ports entrants sur les instances ou autoriser les connexions à des instances sans adresse IP publique.
- Les administrateurs qui veulent accorder et révoquer des accès à partir d'un point central, et fournir aux utilisateurs une solution unique pour les instances Windows et Linux.
- Les utilisateurs finaux qui souhaitent se connecter à une instance en un seul clic depuis le navigateur ou AWS CLI sans avoir à fournir de clés SSH.

Quelles sont les principales fonctionnalités Session Manager ?

- Prise en charge des instances Windows Server et Linux

  Session Manager vous permet d'établir des connexions sécurisées à vos instances Amazon Elastic Compute Cloud (EC2), instances sur site et machines virtuelles (VM). Pour obtenir une liste des types de systèmes d'exploitation Windows et Linux pris en charge, consultez Configuration d'Session Manager (p. 873).

  > **Note**
  >
  > La prise en charge de Session Manager pour les serveurs sur site est assurée pour le niveau des instances avancées uniquement. Pour de plus amples informations, veuillez consulter Activation du niveau d'instances avancées (p. 853).

- Accès aux Session Manager fonctionnalités de la console, de l'interface de ligne de commande et du kit SDK

  Vous pouvez utiliser les Session Manager de l'une des façons suivantes :

  La AWS Systems Manager console inclut l'accès à toutes les Session Manager fonctionnalités pour les administrateurs et les utilisateurs finaux. Vous pouvez effectuer n'importe quelle tâche liée à vos sessions via la console Systems Manager.

  La Amazon EC2 console permet aux utilisateurs finaux de se connecter aux instances EC2 pour lesquelles des autorisations de session leur ont été accordées.

  L'AWS CLI inclut l'accès aux fonctionnalités Session Manager pour les utilisateurs finaux. Vous pouvez démarrer une session, afficher la liste des sessions et mettre fin définitivement à une session à l'aide de l'AWS CLI.

  > **Note**
  >
  > Pour utiliser l'AWS CLI afin d'exécuter des commandes sur votre session, vous devez utiliser la version 1.16.12 de l'interface de ligne de commande (ou une version ultérieure) et vous devez avoir installé le Session Manager plugin sur votre machine locale. Pour plus d'informations, consultez la section (Facultatif) Installer le Session Manager plugin pour l'AWS CLI (p. 917).

  Le Session Manager kit SDK se compose de bibliothèques et d'exemples de code qui permettent aux développeurs d'applications de créer des applications frontales, telles que des shells personnalisés ou des portails en libre-service pour les utilisateurs internes qui utilisent en mode natif Session Manager pour se connecter aux instances . Les développeurs et les partenaires peuvent intégrer Session Manager dans leurs outils côté client ou flux de travail Automation à l'aide d' Session Manager APIs . Vous pouvez même créer des solutions personnalisées.

- IAM contrôle d'accès

  Les stratégies IAM vous permettent de contrôler quels membres de votre organisation peuvent démarrer des sessions sur les instances et à quelles instances ils peuvent accéder. Vous pouvez également

fournir un accès temporaire à vos instances. Par exemple, vous pouvez accorder à un ingénieur de garde (ou à un groupe d'ingénieurs de garde) l'accès aux serveurs de production uniquement pendant la durée de leur service.

- Prise en charge de la capacité de journalisation et d'audit

  Session Manager offre des fonctionnalités d'historique de session de journalisation et d'audit dans votre compte AWS, intégrées à différents autres services AWS. Pour plus d'informations, consultez la section Audit et journalisation de l'activité de session (p. 929).

- Profils shell configurables

  vous Session Manager fournit des options vous permettant de configurer des préférences au sein des sessions. Ces profils personnalisables vous permettent de définir des préférences telles que des préférences shell, des variables d'environnement, des répertoires de travail et l'exécution de plusieurs commandes lorsqu'une session est démarrée.

- Prise en charge du chiffrement des données clés des clients

  Vous pouvez configurer Session Manager pour chiffrer les journaux des données de session dans les journaux que vous envoyez à un compartiment S3 ou diffusez à un groupe de journaux CloudWatch Logs. Vous pouvez également configurer Session Manager pour chiffrer davantage les données transmises entre les ordinateurs clients et vos instances pendant vos sessions. Pour plus d'informations, consultez Audit et journalisation de l'activité de session (p. 929) et Configurer les préférences de session (p. 897).

- AWS PrivateLink pour les instances sans adresse IP publique

  Vous pouvez également configurer des points de terminaison dSystems Manager'un VPC pour utiliser AWS PrivateLink afin de renforcer la sécurité de vos sessions. AWS PrivateLink limite tout le trafic réseau entre vos instances gérées, et Systems Manager au réseau Amazon.Amazon EC2 Pour plus d'informations, consultez la section (Facultatif) Créer un point de terminaison de VPC (p. 42).

- Tunnelisation

  Dans une session, utilisez un document SSM de type session pour tunneliser le trafic, tel que http ou un protocole personnalisé, entre un port local sur un ordinateur client et un port distant sur une instance.

- Des commandes interactives

  Créez un document SSM de type session qui utilise une session pour exécuter de manière interactive une seule commande, ce qui vous permet de gérer les opérations pouvant être effectuées par les utilisateurs sur une instance.

Qu'est-ce qu'une session ?

Une session est une connexion établie à une instance à l'aide de Session Manager. Les sessions sont basées sur un canal de communication bidirectionnel sécurisé entre le client (vous) et l'instance gérée à distance qui diffuse les entrées et les sorties pour les commandes. Le trafic entre un client et une instance gérée est chiffré à l'aide de TLS 1.2, et les demandes de création de la connexion sont signées à l'aide de Sigv4. Cette communication bidirectionnelle permet un accès interactif bash et PowerShell aux instances. Vous pouvez également utiliser une clé AWS Key Management Service (AWS KMS) pour chiffrer davantage les données au-delà du chiffrement TLS par défaut.

Supposons par exemple que John est un ingénieur de garde dans votre service informatique. Il reçoit la notification d'un problème qui lui exige de se connecter à distance à une instance. Il peut s'agir par exemple d'une panne qui nécessite d'être réparée, ou d'une directive pour modifier une option de configuration simple sur une instance. À l'aide de la console AWS Systems Manager, de la console Amazon EC2 ou de l'AWS CLI, John démarre une session qui le connecte à l'instance, exécute les commandes nécessaires sur l'instance afin de réaliser la tâche, puis met fin à la session.

Lorsque John envoie cette première commande pour démarrer la session, le Session Manager service authentifie son ID, vérifie les autorisations qui lui sont accordées par une IAM stratégie , vérifie les paramètres de configuration (par exemple, les limites autorisées pour les sessions) et envoie un message à Agent SSM pour ouvrir la connexion bidirectionnelle. Une fois la connexion établie et après que John ait saisi la commande suivante, le résultat de la commande de l'Agent SSM est chargé vers ce canal de communication et renvoyé vers son ordinateur local.

Rubriques

- Configuration d'Session Manager (p. 873)
- Utilisation des Session Manager (p. 917)
- Audit et journalisation de l'activité de session (p. 929)
- Schéma du document de session (p. 933)
- Dépannage de Session Manager (p. 938)

Configuration d'Session Manager

Avant d'utiliser Session Manager pour vous connecter aux instances Amazon Elastic Compute Cloud (Amazon EC2) de votre compte, suivez les étapes décrites dans les rubriques suivantes.

Rubriques

- Étape 1 : Exécuter les opérations Session Manager prérequises (p. 873)
- Étape 2 : Vérifier ou créer un profil d'instance IAM avec des autorisations Session Manager (p. 876)
- Étape 3 : Contrôler les accès de session utilisateur aux instances (p. 882)
- Étape 4 : Configurer les préférences de session (p. 897)
- Étape 5 : (Facultatif) Restreindre l'accès aux commandes dans une session (p. 908)
- Étape 6 : (Facultatif) Utilisez AWS PrivateLink pour configurer un point de terminaison de VPC pourSession Manager (p. 912)
- Étape 7 : (Facultatif) Désactivez ou activez les autorisations administratives de compte ssm-user (p. 912)
- Étape 8 : (Facultatif) Activez les connexions SSH via Session Manager (p. 915)

Étape 1 : Exécuter les opérations Session Manager prérequises

Avant d'utiliser Session Manager, vérifiez que votre environnement respecte les conditions requises suivantes.

Prérequis d'Session Manager

Exigence	Description
Systèmes d'exploitation pris en charge	AWS Session Manager prend en charge les versions de système d'exploitation suivantes : Note Session Manager prend en charge les instances EC2, ainsi que les serveurs ou machines virtuelles (VM) de votre environnement hybride qui utilisent le niveau instances avancées. Pour de plus amples informations, sur les instances avancées, veuillez consulter Activation du niveau d'instances avancées (p. 853).

Exigence	Description
	Linux Session Manager prend en charge toutes les versions de Linux compatibles avec AWS Systems Manager dans son ensemble. Pour plus d'informations, consultez la section Prérequis d'Systems Manager (p. 15). Windows Session Manager prend en charge Windows Server 2008 R2 via Windows Server 2019. Note Microsoft Windows Server 2016 Nano n'est pas pris en charge.

Exigence	Description
Agent SSM	Au minimum, Agent SSM la version 2.3.68.0 ou une version ultérieure doit être installée sur les instances auxquelles vous souhaitez vous connecter via des sessions. Pour utiliser l'option permettant de chiffrer les données de session à l'aide d'une clé principale client (CMK) créée dans AWS Key Management Service (AWS KMS), la version 2.3.539.0 ou ultérieure d' Agent SSM doit être installée sur l'instance gérée. Pour démarrer un réacheminement de Session Manager port ou une session SSH, la version 2.3.672.0 ou ultérieure d' Agent SSM doit être installée sur l'instance gérée. Pour installer ou mettre à jour Agent SSM, consultez Utilisation des Agent SSM (p. 72). À propos du compte ssm-user A partir de la version 2.3.50.0 d' Agent SSM , l'agent crée un compte utilisateur sur l'instance, avec des privilèges racine ou administrateur, appelé `ssm-user` . (Sur les versions antérieures à 2.3.612.0, le compte est créé au Agent SSM démarrage ou au redémarrage. Sur la version 2.3.612.0 et les versions ultérieures, `ssm-user` est créé la première fois qu'une session commence sur l'instance.) Les sessions sont lancées à l'aide des informations d'identification administratives de ce compte d'utilisateur. Pour en savoir plus sur la restriction de contrôle administrateur pour ce compte, consultez (Facultatif) : Désactivation ou activation des autorisations administrateur du compte ssm-user (p. 912). ssm-user sur les contrôleurs de domaine Windows Server Depuis la version Agent SSM 2.3.612.0, le compte `ssm-user` n'est plus créé automatiquement sur les instances gérées qui sont utilisées en tant que contrôleurs de domaine Windows Server. Pour utiliser le gestionnaire de session sur une machine Windows Server utilisée en tant que contrôleur de domaine, vous devez créer le compte `ssm-user` manuellement s'il n'est pas déjà présent. Sur Windows Server, Agent SSM définit un nouveau mot de passe pour le compte `ssm-user` chaque fois qu'une session commence, ce qui signifie que vous n'avez pas besoin de spécifier un mot de passe lors de la création du compte.

Exigence	Description
AWS CLI	(Facultatif) Si vous utilisez l'AWS CLI pour démarrer vos sessions (au lieu d'utiliser la console AWS Systems Manager ou la console Amazon EC2), la version 1.16.12 ou ultérieure de l'interface de ligne de commande doit être installée sur votre ordinateur local. Vous pouvez appeler aws --version pour vérifier la version. Si vous devez installer ou mettre à niveau l'interface de ligne de commande, consultez la page Installer l'AWS Command Line Interface dans le AWS Command Line Interface Guide de l'utilisateur. **Important** Une version mise à jour de l'Agent SSM est publiée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Si une version antérieure de l'agent est en cours d'exécution sur une instance, certains processus de l'Agent SSM peuvent échouer. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir l'Agent SSM à jour sur vos instances. Pour de plus amples informations, veuillez consulter Automatisation des mises à jour versAgent SSM (p. 109). Pour être alerté des mises à jour de Agent SSM, abonnez-vous à la page Agent SSM Release Notes sur GitHub. En outre, pour utiliser l'interface de ligne de commande afin de gérer vos instances avec Session Manager, vous devez d'abord installer le Session Manager plugin sur votre ordinateur local. Pour plus d'informations, consultez la section (Facultatif) Installer le Session Manager plugin pour l'AWS CLI (p. 917).

Étape 2 : Vérifier ou créer un profil d'instance IAM avec des autorisations Session Manager

Par défaut, AWS Systems Manager n'a pas le droit d'effectuer des actions sur vos instances. Vous devez accorder l'accès en utilisant un profil d'instance AWS Identity and Access Management (IAM). Un profil d'instance est un conteneur qui transmet des informations sur le rôle IAM à une instance Amazon Elastic Compute Cloud (Amazon EC2) lors du lancement. Cette exigence s'applique aux autorisations relatives à toutes les fonctionnalités AWS Systems Manager, et pas uniquement aux fonctionnalités Session Manager.

Si vous utilisez déjà d'autres fonctionnalités Systems Manager, comme Fonctionnalité Exécuter la commande ou Parameter Store, il est possible qu'un profil d'instance avec les autorisations de base requises pour Session Manager soit déjà attaché à vos instances. Si un profil d'instance qui contient la stratégie AWS gérée AmazonSSMManagedInstanceCore est déjà attaché à vos instances , les autorisations requises pour Session Manager sont déjà fournies.

Toutefois, dans certains cas, vous pouvez avoir besoin de modifier les autorisations attachées à votre profil d'instance. Par exemple, vous souhaitez fournir un ensemble plus restreint d'autorisations d'instance, vous avez créé une stratégie personnalisée pour votre profil d'instance ou vous souhaitez utiliser des options de chiffrement Amazon S3 ou AWS KMS pour sécuriser les données de session. Pour ces cas, effectuez l'une des actions suivantes pour autoriser l'exécution d'actions Session Manager sur vos instances :

- Intégrer des autorisations pour Session Manager les actions dans un profil d'instance personnalisé

  Pour ajouter des autorisations pour Session Manager les actions à un profil dIAM'instance existant qui ne s'appuie pas sur la stratégie par défaut fournie par AWS AmazonSSMManagedInstanceCore , suivez les étapes décrites dans .Ajout d'autorisations Session Manager à un profil d'instance existant (p. 877)
- Créer un profil dIAM'instance personnalisé avec Session Manager des autorisations uniquement

  Pour créer un profil d'instance IAM qui contient uniquement des autorisations relatives aux actions Session Manager, suivez les étapes décrites sur la page Créer un profil d'instance IAM personnalisé pour Session Manager (p. 879).
- Crée et utilise un nouveau profil d'instance avec des autorisations pour toutes les Systems Manager actions

  Pour créer un profil d'IAMinstance pour les instances Systems Manager gérées qui utilise une stratégie par défaut fournie par AWS pour accorder toutes les Systems Manager autorisations, suivez les étapes décrites dans Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34) .

  > **Note**
  >
  > Vous pouvez associer un profil d'instance IAM à une instance EC2 lorsque vous la lancez ou à une instance préalablement lancée. Pour de plus amples informations, veuillez consulter la rubrique Profils d'instance.

Rubriques

- Ajout d'autorisations Session Manager à un profil d'instance existant (p. 877)
- Créer un profil d'instance IAM personnalisé pour Session Manager (p. 879)

Ajout d'autorisations Session Manager à un profil d'instance existant

Procédez comme suit pour intégrer Session Manager des autorisations dans un profil d'instance AWS Identity and Access Management (IAM) existant qui ne s'appuie pas sur la stratégie par défaut fournie par AWS AmazonSSMManagedInstanceCore pour les autorisations d'instance . Notez que cette procédure suppose que votre profil existant comprend déjà d'autres autorisations Systems Manager `ssm` relatives aux actions auxquelles vous souhaitez accorder l'accès. Employée seule, cette stratégie ne s'avère pas suffisante pour utiliser Session Manager.

Pour ajouter des autorisations Session Manager à un profil d'instance existant (console)

1. Connectez-vous à la AWS Management Console et ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le volet de navigation, choisissez Roles (Rôles).
3. Dans la liste, sélectionnez le nom du rôle auquel intégrer une stratégie.
4. Choisissez l'onglet Autorisations.

5. Faites défiler jusqu'en bas de la page et choisissez Add inline policy (Ajouter une stratégie en ligne).
6. Choisissez l'onglet JSON.
7. Remplacez le contenu par défaut par ce qui suit :

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssmmessages:CreateControlChannel",
                "ssmmessages:CreateDataChannel",
                "ssmmessages:OpenControlChannel",
                "ssmmessages:OpenDataChannel"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetEncryptionConfiguration"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "key-name"
        }
    ]
}
```

À propos de « ssmmessages »

Pour de plus amples informations sur `ssmmessages`, veuillez consulter Référence : ec2messages, ssmmessages et autres appels d'API (p. 1399).

À propos de « kms : Decrypt »

Dans cette stratégie, l'autorisation `kms:Decrypt` active les clients de chiffrement et de déchiffrement clés pour les données de session. Si vous utilisez le chiffrement AWS Key Management Service (AWS KMS) pour vos données de session, remplacez *`key-name`* avec l'Amazon Resource Name (ARN) de la clé principale client (CMK) que vous souhaitez utiliser, au format `arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-12345EXAMPLE` .

Si vous n'avez pas l'intention d'utiliser AWS KMS le chiffrement pour vos données de session, vous pouvez supprimer le contenu suivant de la stratégie.

```
,
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "key-name"
        }
```

Pour de plus amples informations sur l'utilisation de AWS KMS et une clé CMK pour chiffrer les données de session, veuillez consulter Activer le chiffrement de clé AWS KMS des données de session (console) (p. 902).

8. Choisissez Examiner une stratégie.
9. Sur la page Examiner une stratégie, dans le champ Nom, saisissez un nom pour la stratégie en ligne, tel que **`SessionManagerPermissions`**.
10. Choisissez Create policy (Créer une stratégie).

Créer un profil d'instance IAM personnalisé pour Session Manager

Vous pouvez créer un profil d'instance personnalisé AWS Identity and Access Management (IAM) qui autorise uniquement des Session Manager actions sur vos instances . Vous pouvez également créer une stratégie pour fournir les autorisations nécessaires afin que les journaux d'activité de session soient envoyés à Amazon S3 et CloudWatch Logs.

Une fois que vous avez créé un profil d'instance, veuillez consulter Attacher un rôle IAM à une instance et Attacher ou remplacer un profil d'instance pour découvrir comment attacher un profil d'instance à une instance. Pour en savoir plus sur les rôles et les profils d'instance IAM, veuillez consulter Utilisation de profils d'instance et Rôles IAM pour Amazon EC2 dans le IAM Guide de l'utilisateur.

Rubriques

- Création d'un profil d'instance avec les autorisations Session Manager minimales (console) (p. 879)
- Création d'un profil d'instance avec les autorisations pour Session Manager, Amazon S3 et CloudWatch Logs (p. 881)

Création d'un profil d'instance avec les autorisations Session Manager minimales (console)

Utilisez la procédure suivante pour créer un profil dIAM'instance personnalisé avec une stratégie qui fournit des autorisations uniquement pour Session Manager les actions sur vos instances.

Créer un profil d'instance avec des autorisations Session Manager minimales

1. Connectez-vous à la AWS Management Console et ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation, choisissez Stratégies, puis Créer une stratégie. (Si un bouton Mise en route est affiché, choisissez-le, puis choisissez Créer une stratégie.)
3. Choisissez l'onglet JSON.
4. Remplacez le contenu par défaut par ce qui suit :

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:UpdateInstanceInformation",
                "ssmmessages:CreateControlChannel",
                "ssmmessages:CreateDataChannel",
                "ssmmessages:OpenControlChannel",
                "ssmmessages:OpenDataChannel"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
```

```
            "Action": [
                "s3:GetEncryptionConfiguration"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "key-name"
        }
    ]
}
```

À propos de « ssmmessages »

Pour de plus amples informations sur `ssmmessages`, veuillez consulter Référence : ec2messages, ssmmessages et autres appels d'API (p. 1399).

À propos de « kms : Decrypt »

Dans cette stratégie, l'autorisation `kms:Decrypt` active les clients de chiffrement et de déchiffrement clés pour les données de session. Si vous utilisez le chiffrement AWS Key Management Service (AWS KMS) pour vos données de session, remplacez *`key-name`* avec l'Amazon Resource Name (ARN) de la clé principale client (CMK) que vous souhaitez utiliser, au format `arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-12345EXAMPLE` .

Si vous n'avez pas l'intention d'utiliser AWS KMS le chiffrement pour vos données de session, vous pouvez supprimer le contenu suivant de la stratégie.

```
,
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "key-name"
        }
```

Pour de plus amples informations sur l'utilisation de AWS KMS et une clé CMK pour chiffrer les données de session, veuillez consulter Activer le chiffrement de clé AWS KMS des données de session (console) (p. 902).

5. Choisissez Examiner une stratégie.
6. Sur la page Examiner une stratégie, dans le champ Nom, saisissez un nom pour la stratégie en ligne, tel que **`SessionManagerPermissions`**.
7. (Facultatif) Dans le champ Description, saisissez une description pour la stratégie.
8. Choisissez Create policy (Créer une stratégie).
9. Dans le volet de navigation, choisissez Rôles, puis Créer un rôle.
10. Sur la page Créer un rôle, sélectionnez Service AWS, puis dans la liste Choisir le service qui utilisera ce rôle, sélectionnez EC2.
11. Choisissez Next (Suivant) : Autorisations.
12. Sur la page Attached permissions policy (Stratégie d'autorisations attachée), cochez la case située à gauche du nom de la stratégie que vous venez de créer, comme **`SessionManagerPermissions`**.
13. Choisissez Next (Suivant) : Passez en revue.
14. Sur la page Review (Vérifier), pour Nom du rôle, saisissez un nom pour le profil d'instance IAM, tel que **`MySessionManagerInstanceProfile`**.

15. (Facultatif) Dans le champ Description du rôle, saisissez une description pour le profil d'instance.
16. Sélectionnez Créer un rôle.

Création d'un profil d'instance avec les autorisations pour Session Manager, Amazon S3 et CloudWatch Logs

Utilisez la procédure suivante pour créer un profil dIAM'instance personnalisé avec une stratégie qui fournit des autorisations pour les Session Manager actions sur vos instances. La stratégie fournit également les autorisations nécessaires pour que les journaux de session soient stockés dans des compartiments Amazon Simple Storage Service (Amazon S3) et des groupes Amazon CloudWatch Logs de journaux .

Pour en savoir plus sur la spécification des préférences de stockage des journaux de session, consultez Audit et journalisation de l'activité de session (p. 929).

Créer un profil d'instance autorisant les actions de journalisation pour Session Manager, Amazon S3 et CloudWatch Logs

1. Connectez-vous à la AWS Management Console et ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation, choisissez Stratégies, puis Créer une stratégie. (Si un bouton Mise en route est affiché, choisissez-le, puis choisissez Créer une stratégie.)
3. Choisissez l'onglet JSON.
4. Remplacez le contenu par défaut par la commande suivante : Veillez à remplacer *DOC-EXAMPLE-BUCKET* et *s3-bucket-prefix* avec les noms de votre compartiment et son préfixe (le cas échéant). Pour de plus amples informations sur `ssmmessages` dans la stratégie suivante, veuillez consulter Référence : ec2messages, ssmmessages et autres appels d'API (p. 1399).

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssmmessages:CreateControlChannel",
                "ssmmessages:CreateDataChannel",
                "ssmmessages:OpenControlChannel",
                "ssmmessages:OpenDataChannel",
                "ssm:UpdateInstanceInformation"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogStream",
                "logs:PutLogEvents",
                "logs:DescribeLogGroups",
                "logs:DescribeLogStreams"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:PutObject"
            ],
            "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET/s3-bucket-prefix"
        },
        {
            "Effect": "Allow",
```

```
            "Action": [
                "s3:GetEncryptionConfiguration"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "kms:GenerateDataKey",
            "Resource": "*"
        }
    ]
}
```

> **Important**
>
> Pour générer les journaux de session dans un compartiment S3 appartenant à un autre AWS compte, vous devez ajouter l'IAM`s3:PutObjectAcl`autorisation à cette stratégie. Si cette autorisation n'est pas ajoutée, le compte qui possède le compartiment S3 ne peut pas accéder à la session de la sortie des journaux.

5. Choisissez Examiner une stratégie.
6. Sur la page Examiner une stratégie, dans le champ Nom, saisissez un nom pour la stratégie en ligne, tel que **`SessionManagerPermissions`**.
7. (Facultatif) Dans le champ Description, saisissez une description pour la stratégie.
8. Choisissez Create policy (Créer une stratégie).
9. Dans le volet de navigation, choisissez Rôles, puis Créer un rôle.
10. Sur la page Créer un rôle, sélectionnez Service AWS, puis dans la liste Choisir le service qui utilisera ce rôle, sélectionnez EC2.
11. Choisissez Next (Suivant) : Autorisations.
12. Sur la page Attached permissions policy (Stratégie d'autorisations attachée), cochez la case située à gauche du nom de la stratégie que vous venez de créer, comme **`SessionManagerPermissions`**.
13. Choisissez Next (Suivant) : Passez en revue.
14. Sur la page Review (Vérifier), pour Nom du rôle, saisissez un nom pour le profil d'instance IAM, tel que **`MySessionManagerInstanceProfile`**.
15. (Facultatif) Dans le champ Description du rôle, saisissez une description pour le profil d'instance.
16. Sélectionnez Créer un rôle.

Étape 3 : Contrôler les accès de session utilisateur aux instances

Session Manager vous permet d'accorder et de révoquer les accès utilisateur aux instances de manière centralisée. À l'aide de stratégies AWS Identity and Access Management (IAM), vous contrôlez les instances auxquelles des utilisateurs ou des groupes spécifiques peuvent se connecter, ainsi que les actions Session Manager d'API auxquelles ils peuvent exécuter sur les instances auxquelles ils bénéficient d'un accès.

À propos des formats des ARN d'ID de session

IAMLes stratégies pour Session Manager l'accès utilisent des variables pour les noms d'utilisateur dans le cadre de la session IDs . Les sessions sont ensuite utilisées dans les ARN (Amazon Resource Name) de session pour contrôler l'accès.IDs Les sessions ARNs ont le format suivant :

```
arn:aws:ssm:region-id:account-id:session/session-id
```

Exemple :

```
arn:aws:ssm:us-east-2:123456789012:session/JohnDoe-1a2b3c4d5eEXAMPLE
```

Vous pouvez utiliser une paire de IAM stratégies par défaut fournies par AWS , une pour les utilisateurs finaux et une pour les administrateurs, afin de fournir des autorisations pour les Session Manager activités . Vous pouvez également créer des stratégies IAM personnalisées qui sauront mieux répondre à vos exigences en matière d'autorisations.

Pour plus d'informations sur l'utilisation IAM des variables dans les stratégies, consultez Éléments IAM de stratégie : Variables.

Pour plus d'informations sur la création de stratégies et leur liaison avec des utilisateurs ou des groupes IAM, consultez Création de stratégies IAM et Ajout et suppression de stratégies IAM dans le IAM Guide de l'utilisateur.

Rubriques

- Appliquer une vérification des autorisations de document de session pour AWS CLI (p. 883)
- Démarrage rapide - Stratégies IAM par défaut pour Session Manager (p. 885)
- Exemples de stratégies IAM supplémentaires pour Session Manager (p. 891)

Appliquer une vérification des autorisations de document de session pour AWS CLI

Lorsque vous configurez Session Manager pour votre compte, le système crée un `Session` document de type SSM nommé `SSM-SessionManagerRunShell` . Ce SSM document stocke vos préférences de session, par exemple si les données de session sont enregistrées dans un compartiment Amazon Simple Storage Service (Amazon S3) ou un groupe de Amazon CloudWatch Logs journaux , si les données de session sont chiffrées à l'aide d' et si la prise en charge de Run As est activée pour vos sessions.AWS Key Management Service Voici un exemple pour .

```
{
  "schemaVersion": "1.0",
  "description": "Document to hold regional settings for Session Manager",
  "sessionType": "Standard_Stream",
  "inputs": {
    "s3BucketName": "DOC-EXAMPLE-BUCKET",
    "s3KeyPrefix": "MyBucketPrefix",
    "s3EncryptionEnabled": true,
    "cloudWatchLogGroupName": "MyLogGroupName",
    "cloudWatchEncryptionEnabled": true,
    "kmsKeyId": "MyKMSKeyID",
    "runAsEnabled": true,
    "runAsDefaultUser": "MyDefaultRunAsUser"
  }
}
```

Par défaut, si un utilisateur de votre compte a reçu dans sa stratégie d'utilisateur AWS Identity and Access Management (IAM) l'autorisation de démarrer des sessions, cet utilisateur a accès au `SSM-SessionManagerRunShell` SSM document. Cela signifie que lorsqu'il utilise AWS CLI pour exécuter la commande `start-session` et qu'il ne spécifie pas de document dans l'option `--document-name`, le système utilise `SSM-SessionManagerRunShell` et lance la session. La session démarre, même si la stratégie IAM de l'utilisateur n'accorde pas explicitement l'autorisation d'accéder au document `SSM-SessionManagerRunShell`.

Par exemple, la commande suivante ne spécifie pas de document de session.

```
aws ssm start-session \
```

```
    --target i-02573cafcfEXAMPLE
```

L'exemple suivant spécifie le document de session Session Manager par défaut.

```
aws ssm start-session \
    --document-name SSM-SessionManagerRunShell \
    --target i-02573cafcfEXAMPLE
```

Pour restreindre l'accès au document de session par défaut ou à n'importe quel document de session, vous pouvez ajouter un élément de condition à la stratégie IAM de l'utilisateur qui confirme si l'utilisateur dispose d'un accès explicite à un document de session. Lorsque cette condition est appliquée, l'utilisateur doit spécifier une valeur pour l'option `--document-name` de la commande AWS CLI `start-session`. Cette valeur correspond au document de session Session Manager par défaut ou à un document de session personnalisé que vous avez créé. L'élément de condition suivant, lorsqu'il est ajouté à l'action `ssm:StartSession` dans la stratégie IAM, procède à une vérification de l'accès au document de session.

```
"Condition": {
    "BoolIfExists": {
        "ssm:SessionDocumentAccessCheck": "true"
    }
}
```

Avec cet élément de condition défini sur `true` , un accès explicite à un document de session doit être accordé dans la IAM stratégie pour que l'utilisateur puisse démarrer une session. Voici un exemple pour .

```
{
    "Effect": "Allow",
    "Action": [
        "ssm:StartSession"
    ],
    "Resource": [
        "arn:aws:ec2:*:*:instance/instance-id",
        "arn:aws:ssm:region:account-id:document/SSM-SessionManagerRunShell"
    ]
}
```

Pour obtenir un exemple de spécification d'un document de session Session Manager dans une stratégie IAM, veuillez consulter Démarrage rapide - Stratégies d'utilisateur final pour Session Manager (p. 885).

Autres scénarios

L'utilisation du document de session `SSM-SessionManagerRunShell` par défaut est le seul cas où un nom de document peut être omis dans la commande `start-session` de l'interface de ligne de commande. Dans les autres cas, l'utilisateur doit spécifier une valeur pour l'option `--document-name` de la commande AWS CLI `start-session`. Le système vérifie si l'utilisateur dispose d'un accès explicite au document de session qu'il spécifie.

Par exemple, si un utilisateur spécifie le nom d'un document de session personnalisé que vous avez créé, la stratégie IAM de l'utilisateur doit lui accorder l'autorisation d'accéder à ce document.

Si un utilisateur exécute une commande pour démarrer une session à l'aide de SSH, la stratégie de l'utilisateur doit lui accorder l'accès au document de session `AWS-StartSSHSession`.

> **Note**
>
> Pour démarrer une session à l'aide de SSH, les étapes de configuration doivent être effectuées sur l'instance cible et sur l'ordinateur local de l'utilisateur. Pour plus d'informations, consultez la section (Facultatif) Activez les connexions SSH via Session Manager (p. 915).

Démarrage rapide - Stratégies IAM par défaut pour Session Manager

Aidez-vous des exemples de cette section pour créer des stratégies AWS Identity and Access Management (IAM) qui fournissent les autorisations d'Session Manageraccès les plus couramment nécessaires.

> **Note**
>
> Vous pouvez également utiliser une stratégie de clé AWS Key Management Service (AWS KMS) pour contrôler IAM les utilisateurs, IAM les rôles et les AWS comptes qui bénéficient d'un accès à votre clé principale client (CMK). Pour de plus amples informations, veuillez consulter Présentation de la gestion de l'accès à vos ressources AWS KMS et Utilisation de stratégies de clé dans AWS KMS le AWS Key Management Service Developer Guide.

Rubriques

- Démarrage rapide - Stratégies d'utilisateur final pour Session Manager (p. 885)
- Démarrage rapide - Stratégie d'administrateur pour Session Manager (p. 888)

Démarrage rapide - Stratégies d'utilisateur final pour Session Manager

Utilisez les exemples suivants pour créer des stratégies d'utilisateur final IAM pour Session Manager.

Vous pouvez créer une stratégie qui permet aux utilisateurs de démarrer des sessions uniquement à partir de la Session Manager console et AWS Command Line Interface de l' (AWS CLI), uniquement à partir de la console Amazon Elastic Compute Cloud (Amazon EC2), ou à partir des trois.

Ces stratégies permettent aux utilisateurs finaux de démarrer une session sur une instance particulière et de mettre fin à leurs propres sessions uniquement. Consultez la page Exemples de stratégies IAM supplémentaires pour Session Manager (p. 891) pour obtenir des exemples de personnalisation de la stratégie.

> **Note**
>
> Dans tous les exemples de stratégies suivants, remplacez *instance-id* avec l'ID de l'instance à laquelle vous souhaitez accorder l'accès, au format `i-02573cafcfEXAMPLE` . Remplacez *region* et *account-id* par votre AWS région et votre ID de compte AWS, tels que `us-east-2` et `111122223333` .

Reportez-vous aux sections suivantes pour voir des exemples de stratégies pour la plage d'accès aux sessions que vous souhaitez fournir.

Session Manager and CLI

Utilisez cet exemple de stratégie pour permettre aux utilisateurs de démarrer des sessions uniquement à partir de la console Session Manager et de l'AWS CLI. Cette stratégie ne fournit pas toutes les autorisations nécessaires pour démarrer des sessions à partir de la console Amazon EC2.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:instance/instance-id",
                "arn:aws:ssm:region:account-id:document/SSM-SessionManagerRunShell"  (1)
            ],
            "Condition": {
                "BoolIfExists": {
```

```
                    "ssm:SessionDocumentAccessCheck": "true" 2
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeSessions",
                "ssm:GetConnectionStatus",
                "ssm:DescribeInstanceProperties",
                "ec2:DescribeInstances"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:username}-*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:GenerateDataKey" 3
            ],
            "Resource": "key-name"
        }
    ]
}
```

Amazon EC2

Utilisez cet exemple de stratégie pour permettre aux utilisateurs de démarrer des sessions uniquement à partir de la console Amazon EC2. Cette stratégie ne fournit pas toutes les autorisations nécessaires pour démarrer des sessions à partir de la console Session Manager et de l'AWS CLI.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession",
                "ssm:SendCommand" 4
            ],
            "Resource": [
                "arn:aws:ec2:*:*:instance/instance-id"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetConnectionStatus",
                "ssm:DescribeInstanceInformation"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
```

```
                "ssm:TerminateSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:username}-*"
            ]
        }
    ]
}
```

Session Manager, CLI, and Amazon EC2

Utilisez cet exemple de stratégie pour permettre aux utilisateurs de démarrer des sessions à partir de la console Session Manager, de l'AWS CLI et de la console Amazon EC2.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession",
                "ssm:SendCommand"  4
            ],
            "Resource": [
                "arn:aws:ec2:*:*:instance/instance-id",
                "arn:aws:ssm:region:account-id:document/SSM-SessionManagerRunShell"  1
            ],
            "Condition": {
                "BoolIfExists": {
                    "ssm:SessionDocumentAccessCheck": "true"  2
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeSessions",
                "ssm:GetConnectionStatus",
                "ssm:DescribeInstanceInformation",
                "ssm:DescribeInstanceProperties",
                "ec2:DescribeInstances"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:username}-*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:GenerateDataKey"  3
            ],
            "Resource": "key-name"
        }
    ]
}
```

[1] `SSM-SessionManagerRunShell` est le nom par défaut du document SSM créé par Session Manager pour stocker vos préférences de configuration de session. Vous pouvez créer un document de session personnalisé et le spécifier dans cette stratégie à la place. Vous pouvez également spécifier le document fourni par AWS pour les utilisateurs qui démarrent des sessions à l'aide de SSH.`AWS-StartSSHSession` Pour plus d'informations sur les étapes de configuration nécessaires pour prendre en charge les sessions à l'aide de SSH, consultez (Facultatif) Activez les connexions SSH via Session Manager (p. 915).

[2] Si vous définissez l'élément de condition `ssm:SessionDocumentAccessCheck` sur `true`, le système vérifie qu'un utilisateur dispose d'un accès explicite au document de session défini (`SSM-SessionManagerRunShell` dans cet exemple) avant d'établir une session. Pour plus d'informations, consultez la section Appliquer une vérification des autorisations de document de session pour AWS CLI (p. 883).

[3] L'autorisation `kms:GenerateDataKey` permet la création d'une clé de chiffrement des données qui sera utilisée pour chiffrer les données de session. Si vous utilisez le chiffrement AWS Key Management Service (AWS KMS) pour vos données de session, remplacez *`key-name`* avec l'Amazon Resource Name (ARN) de la clé principale client (CMK) que vous souhaitez utiliser, au format `arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-12345EXAMPLE` . Si vous n'utilisez pas le chiffrement AWS KMS de clé pour vos données de session, supprimez le contenu suivant de la stratégie.

```
,
        {
            "Effect": "Allow",
            "Action": [
                "kms:GenerateDataKey"
            ],
            "Resource": "key-name"
        }
```

Pour plus d'informations sur AWS KMS et CMKs pour le chiffrement des données de session, consultez Activer le chiffrement de clé AWS KMS des données de session (console) (p. 902) .

[4] L'autorisation pour SendCommand est nécessaire dans les cas où un utilisateur tente de démarrer une session à partir de la console Amazon EC2, mais une commande doit d'abord être envoyée pour mettre à jour l'Agent SSM.

Démarrage rapide - Stratégie d'administrateur pour Session Manager

Utilisez les exemples suivants pour créer des stratégies d'administrateur IAM pour Session Manager.

Ces stratégies permettent aux administrateurs de démarrer une session sur les instances qui sont balisées avec `Key=Finance,Value=WebServers` , de créer, mettre à jour et supprimer des préférences, et de mettre fin à leurs propres sessions uniquement. Consultez la page Exemples de stratégies IAM supplémentaires pour Session Manager (p. 891) pour obtenir des exemples de personnalisation de la stratégie.

Vous pouvez créer une stratégie qui autorise les administrateurs à effectuer ces tâches uniquement à partir de la console Session Manager et de l'AWS CLI, uniquement à partir de la console Amazon EC2, ou à partir des trois.

> **Note**
>
> Remplacez *`region`* et *`account-id`* par votre AWS région et votre ID de compte AWS, tels que `us-east-2` et `111122223333` .

Reportez-vous aux sections suivantes pour voir des exemples de stratégies pour les trois scénarios d'autorisations.

Session Manager and CLI

Utilisez cet exemple de stratégie pour permettre aux administrateurs d'effectuer les tâches liées à la session uniquement à partir de la console Session Manager et de l'AWS CLI. Cette stratégie ne fournit pas toutes les autorisations nécessaires pour effectuer les tâches liées à la session à partir de la console Amazon EC2.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:instance/*"
            ],
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/Finance": [
                        "WebServers"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeSessions",
                "ssm:GetConnectionStatus",
                "ssm:DescribeInstanceProperties",
                "ec2:DescribeInstances"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:CreateDocument",
                "ssm:UpdateDocument",
                "ssm:GetDocument"
            ],
            "Resource": "arn:aws:ssm:region:account-id:document/SSM-
SessionManagerRunShell"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:username}-*"
            ]
        }
    ]
}
```

Amazon EC2

Utilisez cet exemple de stratégie pour permettre aux administrateurs d'effectuer les tâches liées à la session uniquement à partir de la Amazon EC2 console . Cette stratégie ne fournit pas toutes les autorisations nécessaires pour effectuer les tâches liées à la session à partir de la console Session Manager et de l'AWS CLI.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession",
                "ssm:SendCommand"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:instance/*"
            ],
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/tag-key": [
                        "tag-value"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetConnectionStatus",
                "ssm:DescribeInstanceInformation"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:username}-*"
            ]
        }
    ]
}
```

Session Manager, CLI, and Amazon EC2

Utilisez cet exemple de stratégie pour permettre aux administrateurs d'effectuer les tâches liées à la session à partir de la console Session Manager, de l'AWS CLI et de la console Amazon EC2.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession",
                "ssm:SendCommand"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:instance/*"
            ],
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/tag-key": [
                        "tag-value"
                    ]
                }
```

```
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeSessions",
                "ssm:GetConnectionStatus",
                "ssm:DescribeInstanceInformation",
                "ssm:DescribeInstanceProperties",
                "ec2:DescribeInstances"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:CreateDocument",
                "ssm:UpdateDocument",
                "ssm:GetDocument"
            ],
            "Resource": "arn:aws:ssm:region:account-id:document/SSM-
SessionManagerRunShell"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:username}-*"
            ]
        }
    ]
}
```

[1] L'autorisation pour SendCommand est nécessaire dans les cas où un utilisateur tente de démarrer une session à partir de la console Amazon EC2, mais une commande doit d'abord être envoyée pour mettre à jour l'Agent SSM.

Exemples de stratégies IAM supplémentaires pour Session Manager

Aidez-vous des exemples de stratégies suivants pour vous aider à créer une stratégie AWS Identity and Access Management (IAM) personnalisée pour les scénarios Session Manager d'accès utilisateur que vous souhaitez prendre en charge.

Rubriques

- Exemple 1 : Restreindre l'accès de la fonctionnalité à des instances spécifiques (p. 891)
- Exemple 2 : Restreindre l'accès en fonction des balises d'instance (p. 892)
- Exemple 3 : Autoriser un utilisateur à mettre fin uniquement aux sessions qu'il a démarrées (p. 893)
- Exemple 4 : Autoriser un accès complet (administrateur) à toutes les sessions (p. 896)

Exemple 1 : Restreindre l'accès de la fonctionnalité à des instances spécifiques

Vous pouvez restreindre l'accès à des instances spécifiques en créant une stratégie IAM utilisateur qui inclut IIDs' des instances. Dans l'exemple suivant, l'utilisateur détient une autorisation Session Manager pour accéder à trois instances spécifiques uniquement, et à mettre fin uniquement à ses sessions sur ces instances. Si l'utilisateur envoie une commande à une autre instance ou tente de mettre fin à une autre session, le résultat de la commande affiche `AccessDenied`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-2:123456789012:instance/i-1234567890EXAMPLE",
                "arn:aws:ec2:us-east-2:123456789012:instance/i-abcdefghijEXAMPLE",
                "arn:aws:ec2:us-east-2:123456789012:instance/i-0e9d8c7b6aEXAMPLE"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:username}-*"
            ]
        }
    ]
}
```

Exemple 2 : Restreindre l'accès en fonction des balises d'instance

Vous pouvez restreindre l'accès à des instances en fonction de balises Amazon EC2 spécifiques. Dans l'exemple suivant, l'utilisateur est autorisé à démarrer des sessions (`Effect: Allow, Action: ssm:StartSession`) sur n'importe quelle instance (`Resource: arn:aws:ec2:*:*:instance/*`) à condition que l'instance soit une Finance WebServer (`ssm:resourceTag/Finance: WebServer`). Si l'utilisateur envoie une commande à une instance non balisée ou qui possède une balise autre que `Finance: WebServer`, le résultat de la commande affiche `AccessDenied`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:instance/*"
            ],
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/Finance": [
                        "WebServers"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:username}-*"
            ]
        }
```

```
    ]
}
```

Vous pouvez créer des stratégies IAM qui permettent à un utilisateur de démarrer des sessions sur des instances qui contiennent plusieurs balises. La stratégie suivante permet à l'utilisateur de démarrer des sessions sur des instances auxquelles les deux balises spécifiées sont appliquées. Si un utilisateur envoie une commande à une instance qui ne contient pas ces balises, le résultat de la commande affiche `AccessDenied`.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "ssm:StartSession"
         ],
         "Resource":"*",
         "Condition":{
            "StringLike":{
               "ssm:resourceTag/tag-key1":[
                  "tag-value1"
               ],
               "ssm:resourceTag/tag-key2":[
                  "tag-value2"
               ]
            }
         }
      }
   ]
}
```

Pour de plus amples informations sur la création des stratégies utilisateur IAM, veuillez consulter Stratégies gérées et stratégies en ligne dans le IAM Guide de l'utilisateur. Pour de plus amples informations sur le balisage des instances, veuillez consulter Balisage de vos ressources Amazon EC2 dans le Amazon EC2 Guide de l'utilisateur pour les instances Linux (le contenu s'applique aux instances Windows et Linux). Pour de plus amples informations sur la manière de renforcer votre sécurité contre des commandes de niveau racine non autorisées sur vos instances, veuillez consulter la page Restriction de l'accès aux commandes de niveau racine viaAgent SSM (p. 108).

Exemple 3 : Autoriser un utilisateur à mettre fin uniquement aux sessions qu'il a démarrées

Session Managerfournit deux méthodes pour contrôler les sessions auxquelles un utilisateur de votre AWS compte est autorisé à se terminer.

- Utilisez la variable `{aws:username}` dans une stratégie d'autorisations AWS Identity and Access Management (IAM). Les utilisateurs ne peuvent terminer que les sessions qu'ils ont démarrées. Cette méthode ne fonctionne pas pour les comptes qui utilisent la fédération IDs pour accorder l'accès à AWS . Les fédérés IDs utilisent la variable `{aws:userid}` au lieu de `{aws:username}` .
- Utilisez les balises fournies par AWS les balises dans une stratégie dIAM'autorisations . Dans la stratégie, vous incluez une condition qui autorise les utilisateurs à mettre fin uniquement aux sessions balisées avec des balises spécifiques qui ont été fournies par AWS . Cette méthode fonctionne pour tous les comptes, y compris ceux qui utilisent la fédération IDs pour accorder l'accès à AWS .

Méthode 1 : Accorder TerminateSession des privilèges à l'aide de la variable`{aws:username}`

La IAM stratégie suivante permet à un utilisateur d'afficher toutes les sessions de votre compte.IDs Toutefois, les utilisateurs ne peuvent interagir avec les instances que par le biais des sessions qu'ils ont

démarrées. Un utilisateur auquel vous attribuez la stratégie suivante ne peut pas se connecter ni mettre fin aux sessions des autres utilisateurs. Cette stratégie utilise la variable `{aws:username}`.

> **Note**
>
> Cette méthode ne fonctionne pas pour les comptes qui accordent l'accès à à AWS l'aide de IDs fédéré.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "ssm:DescribeSessions"
            ],
            "Effect": "Allow",
            "Resource": [
                "*"
            ]
        },
        {
            "Action": [
                "ssm:TerminateSession"
            ],
            "Effect": "Allow",
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:username}-*"
            ]
        }
    ]
}
```

Méthode 2 : Accorder TerminateSession des privilèges à l'aide des balises fournies parAWS

Vous pouvez contrôler les sessions auxquelles un utilisateur peut mettre fin à l'aide d'une condition avec des variables de clé de balise spécifiques dans une stratégie IAM utilisateur . La condition spécifie que l'utilisateur ne peut terminer que les sessions qui sont marquées avec une ou deux des variables de clé de balise spécifiques et une valeur spécifiée.

Lorsqu'un utilisateur de votre AWS compte démarre une session, Session Manager applique deux balises de ressources à la session. La première balise de ressource est `aws:ssmmessages:target-id`, avec laquelle vous spécifiez l'ID de la cible à laquelle l'utilisateur est autorisé à se terminer. L'autre balise de ressource est `aws:ssmmessages:session-id`, avec une valeur au format *`role-id`*`:`*`caller-specified-role-name`*.

> **Note**
>
> Session Manager ne prend pas en charge les balises personnalisées pour cette stratégie de contrôle d'accès IAM. Vous devez utiliser les balises de ressource fournies par AWS , décrites ci-dessous.

aws:ssmmessages:target-id

Avec cette clé de balise, vous incluez l'ID d'instance comme valeur dans la stratégie. Dans le bloc de stratégie suivant, l'instruction de condition permet à un utilisateur de mettre fin uniquement à l'instance i-02573cafcfEXAMPLE .

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

```
            "Action": [
               "ssm:TerminateSession"
            ],
            "Resource": "",
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/aws:ssmmessages:target-id": [
                       "i-02573cafcfEXAMPLE"
                    ]
                }
            }
        }
    ]
}
```

Si l'utilisateur tente de mettre fin à une session pour laquelle il n'a pas obtenu cette autorisation `TerminateSession`, il reçoit une `AccessDeniedException` erreur.

aws:ssmmessages:session-id

Cette clé de balise inclut une variable pour l'ID de session comme valeur dans la demande de démarrage d'une session.

L'exemple suivant illustre une stratégie pour les cas où le type d'appelant est `User` . La valeur que vous fournissez pour `aws:ssmmessages:session-id` est l'ID de l'utilisateur. Dans cet exemple, `AIDIODR4TAW7CSEXAMPLE` représente l'ID d'un utilisateur dans votre AWS compte. Pour récupérer l'ID d'un utilisateur dans votre AWS compte, utilisez la IAM commande `get-user` , . Pour plus d'informations, consultez get-user dans la AWS Identity and Access Management section du IAM Guide de l'utilisateur .

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
               "ssm:TerminateSession"
            ],
            "Resource": "",
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/aws:ssmmessages:session-id": [
                       "AIDIODR4TAW7CSEXAMPLE"
                    ]
                }
            }
        }
    ]
}
```

L'exemple suivant illustre une stratégie pour les cas où le type d'appelant est `AssumedRole` . Vous pouvez utiliser la `{aws:userid}` variable pour la valeur que vous fournissez pour `aws:ssmmessages:session-id` . Sinon, vous pouvez coder en dur un ID de rôle pour la valeur que vous fournissez pour `aws:ssmmessages:session-id` . Si vous codez en dur un ID de rôle, vous devez fournir la valeur au format *`role-id`*`:`*`caller-specified-role-name`* . Par exemple, `AIDIODR4TAW7CSEXAMPLE:MyRole` .

> **Important**
>
> Pour que les balises système soient appliquées, l'ID de rôle que vous fournissez peut contenir les caractères suivants uniquement : Lettres Unicode, 0-9, espace, _ , . , : , / , = , , + , - et @ .\

Pour récupérer l'ID d'un rôle dans votre AWS compte, utilisez la `get-caller-identity` commande . Pour de plus amples informations, veuillez consulter get-caller-identity dans le AWS CLI Command Reference.

```
{
     "Version": "2012-10-17",
     "Statement": [
         {
             "Effect": "Allow",
             "Action": [
                "ssm:TerminateSession"
             ],
             "Resource": "",
             "Condition": {
                 "StringLike": {
                     "ssm:resourceTag/aws:ssmmessages:session-id": [
                        "${aws:userid}"
                     ]
                 }
             }
         }
     ]
}
```

Si un utilisateur tente de mettre fin à une session pour laquelle il n'a pas obtenu cette autorisation `TerminateSession`, il reçoit une erreur `AccessDeniedException`.

aws:ssmmessages:target-id et aws:ssmmessages:session-id

Vous pouvez également créer IAM des stratégies qui permettent à un utilisateur de mettre fin aux sessions qui sont balisées avec les deux balises système, comme illustré dans cet exemple.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "ssm:TerminateSession"
         ],
         "Resource":"*",
         "Condition":{
            "StringLike":{
               "ssm:resourceTag/aws:ssmmessages:target-id":[
                  "instance-id"
               ],
               "ssm:resourceTag/aws:ssmmessages:session-id":[
                  "${aws:username}-*"
               ]
            }
         }
      }
   ]
}
```

Exemple 4 : Autoriser un accès complet (administrateur) à toutes les sessions

La IAM stratégie suivante permet à un utilisateur d'interagir entièrement avec toutes les instances et toutes les sessions créées par tous les utilisateurs pour toutes les instances. Elle doit être accordée uniquement à un administrateur qui nécessite un contrôle total sur les activités Session Manager de votre organisation.

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "ssm:StartSession",
                "ssm:TerminateSession",
                "ssm:ResumeSession",
                "ssm:DescribeSessions",
                "ssm:GetConnectionStatus"
            ],
            "Effect": "Allow",
            "Resource": [
                "*"
            ]
        }
    ]
}
```

Étape 4 : Configurer les préférences de session

Un utilisateur AWS Identity and Access Management (IAM) disposant d'autorisations d'administrateur peut effectuer les opérations suivantes :

- Activer la prise en charge de Run As pour les instances Linux. Cela permet de démarrer des sessions à l'aide des informations d'identification d'un utilisateur du système d'exploitation spécifié au lieu des informations d'identification d'un compte `ssm-user` généré par le système que Session Manager peut créer sur une instance gérée.
- Configurez Session Manager pour utiliser le chiffrement de clé AWS Key Management Service (AWS KMS) afin de fournir une protection supplémentaire pour les données transmises entre les ordinateurs clients et les instances gérées.
- Configurer Session Manager pour créer et envoyer des journaux d'historique de session à un compartiment Amazon Simple Storage Service (Amazon S3) ou un groupe de journaux Amazon CloudWatch Logs. Les données de journaux stockées peuvent ensuite être utilisées à des fins d'audit ou en tant que rapports sur les connexions à vos instances et les commandes exécutées au cours des sessions.
- Configurez Session Manager pour utiliser des profils shell configurables. Ces profils personnalisables vous permettent de définir des préférences au sein de sessions telles que les préférences shell, les variables d'environnement, les répertoires de travail et l'exécution de plusieurs commandes lorsqu'une session est démarrée.

> **Note**
>
> Avant qu'un utilisateur puisse mettre à jour vos préférences Session Manager, il doit avoir reçu les autorisations spécifiques qui leur permettent d'effectuer ces mises à jour, s'il ne les possède pas déjà. Sans ces autorisations, l'utilisateur ne peut pas configurer les options de journalisation ou définir d'autres préférences de session pour votre compte.

Rubriques

- Accorder ou révoquer des autorisations utilisateur pour mettre à jour des préférences Session Manager (p. 898)
- Activer les profils shell configurables (p. 899)
- Activer la prise en charge de Run As pour les instances Linux (p. 900)
- Activer le chiffrement de clé AWS KMS des données de session (console) (p. 902)
- Créer des préférences Session Manager (ligne de commande) (p. 903)
- Mettre à jour les préférences Session Manager (ligne de commande) (p. 906)

Pour plus d'informations sur l'utilisation Systems Manager de la console pour configurer des options pour la journalisation des données de session, consultez les rubriques suivantes :

- Journalisation des données de session avec Amazon S3 (console) (p. 931)
- Journalisation des données de session avec Amazon CloudWatch Logs (console) (p. 932)

Accorder ou révoquer des autorisations utilisateur pour mettre à jour des préférences Session Manager

Les préférences d'un compte sont stockées sous forme de documents SSM pour chaque région AWS. Pour qu'un utilisateur puisse mettre à jour des préférences de session sur votre compte, il doit détenir les autorisations nécessaires pour accéder au type de document SSM sur lequel ces préférences sont stockées. Ces autorisations sont accordées via une stratégie AWS Identity and Access Management (IAM).

Stratégie administrateur pour autoriser la création et la mise à jour des préférences

Un administrateur peut utiliser la stratégie suivante pour créer et mettre à jour des préférences à tout moment. La stratégie suivante autorise l'accès et la mise à jour du document `SSM-SessionManagerRunShell` sur le compte us-east-2 123456789012.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "ssm:CreateDocument",
                "ssm:UpdateDocument",
                "ssm:DeleteDocument"
            ],
            "Effect": "Allow",
            "Resource": [
                "arn:aws:ssm:us-east-2:123456789012:document/SSM-SessionManagerRunShell"
            ]
        }
    ]
}
```

Stratégie utilisateur pour interdire la mise à jour des préférences

Utilisez la stratégie suivante pour interdire aux utilisateurs finaux de votre compte de mettre à jour ou remplacer des préférences Session Manager.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "ssm:CreateDocument",
                "ssm:UpdateDocument",
                "ssm:DeleteDocument"
            ],
            "Effect": "Deny",
            "Resource": [
                "arn:aws:ssm:us-east-2:123456789012:document/SSM-SessionManagerRunShell"
            ]
        }
    ]
}
```

Activer les profils shell configurables

Par défaut, les sessions sur les instances EC2 pour Linux commencent à utiliser le shell balline (sh). Cependant, vous préférerez peut-être utiliser un autre shell comme bash. En activant des profils shell configurables, vous pouvez personnaliser des préférences au sein de sessions telles que des préférences shell, des variables d'environnement, des répertoires de travail et l'exécution de plusieurs commandes lorsqu'une session est démarrée.

> **Important**
>
> Systems Managerne vérifie pas les commandes ou les scripts de votre profil shell pour voir les modifications qu'ils apporteraient à une instance avant de les exécuter. Pour limiter la capacité d'un utilisateur à modifier des commandes ou des scripts saisis dans son profil shell, nous vous recommandons de procéder comme suit :
>
> - Créez un document personnalisé de type session pour vos IAM utilisateurs et rôles. Modifiez ensuite la IAM stratégie pour ces utilisateurs et rôles afin que l'action `StartSession` d'API puisse uniquement utiliser le document Session-type que vous avez créé pour eux. Pour plus d'informations, consultez Créer des préférences Session Manager (ligne de commande) (p. 903) et Démarrage rapide - Stratégies d'utilisateur final pour Session Manager (p. 885) .
> - Modifiez la IAM stratégie pour vos IAM utilisateurs et rôles pour refuser l'accès à l`UpdateDocument`'action d'API pour la ressource de document de type session que vous créez. Cela permet à vos utilisateurs et rôles d'utiliser le document que vous avez créé pour leurs préférences de session sans leur permettre de modifier les paramètres.

Pour activer les profils shell configurables

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Session Manager.
3. Sélectionnez l'onglet Préférences, puis Modifier.
4. Spécifiez les variables d'environnement, les préférences shell ou les commandes que vous souhaitez exécuter lorsque votre session démarre dans les champs des systèmes d'exploitation applicables.
5. Choisissez Save.

Voici quelques exemples de commandes qui peuvent être ajoutées à votre profil shell.

Passez au shell bash et au répertoire /usr sur les instances Linux.

```
/bin/bash
cd /usr
```

Fournit un horodatage et un message de bienvenue au début d'une session.

Linux

```
timestamp=$(date '+%Y-%m-%dT%H:%M:%SZ')
user=$(whoami)
echo $timestamp && echo "Welcome $user"'!'
echo "You have logged in to a production instance. Note that all session activity is
 being logged."
```

Windows

```
$timestamp = (Get-Date).ToString("yyyy-MM-ddTH:mm:ssZ")
```

```
$splitName = (whoami).Split("\")
$user = $splitName[1]
Write-Host $timestamp
Write-Host "Welcome $user!"
Write-Host "You have logged in to a production instance. Note that all session activity
 is being logged."
```

Afficher l'activité du système dynamique au début d'une session.

Linux

```
top
```

Windows

```
while ($true) { Get-Process | Sort-Object -Descending CPU | Select-Object -First 30; `
Start-Sleep -Seconds 2; cls
Write-Host "Handles  NPM(K)    PM(K)      WS(K) VM(M)   CPU(s)     Id ProcessName";
Write-Host "-------  ------    -----      ----- -----   ------     -- -----------"}
```

Activer la prise en charge de Run As pour les instances Linux

Par défaut, les sessions sont lancées à l'aide des informations d'identification d'un compte `ssm-user` généré par le système qui est créé sur une instance gérée. (Sur les machines Linux, ce compte est ajouté à `/etc/sudoers/` .) Vous pouvez plutôt lancer des sessions à l'aide des informations d'identification d'un compte de système d'exploitation (OS). Session Manager fournit deux méthodes pour spécifier le compte de système d'exploitation à utiliser.

Méthode 1 : Baliser un IAM utilisateur ou un rôle (recommandé)

Vous pouvez spécifier le compte utilisateur du système d'exploitation qui est utilisé pour démarrer des sessions en balisant un utilisateur AWS Identity and Access Management (IAM) ou un rôle associé avec le nom de clé fourni par et en spécifiant le nom d'utilisateur du système d'exploitation comme valeur.AWS`SSMSessionRunAs` Par exemple, si le nom du compte d'utilisateur de système d'exploitation est `DevRoleLogin`, la balise correspondante à utiliser est `SSMSessionRunAs = DevRoleLogin`.

Avec cette méthode, vous pouvez spécifier un nom de compte de système dIAM'exploitation différent pour chaque utilisateur ou rôle que vous balisez, ou utiliser le même nom d'utilisateur de système d'exploitation pour tous les utilisateurs.

Pour plus d'informations sur le balisage IAM des entités, consultez les rubriques suivantes :

- Balisage des entités et objets IAM dans le IAM Guide de l'utilisateur
- Add Tags to Manage Your AWS IAM Users and Roles dans le blog de sécurité AWS

Méthode 2 : Spécifier un nom d'utilisateur de système d'exploitation dans Session Manager les préférences

Lorsque vous configurez Session Manager les préférences dans la console ou à l'aide de l' AWS Command Line Interface (AWS CLI), vous pouvez spécifier le nom d'utilisateur du système d'exploitation avec lequel démarrer les sessions.

Grâce à cette méthode, toutes les sessions sont exécutées par le même utilisateur du système d'exploitation pour tous les IAM utilisateurs de votre compte qui se connectent à l'instance à l'aide d' Session Manager .

Fonctionnement

Si vous activez la prise en charge de Run As pour les sessions, le système vérifie les autorisations d'accès comme suit :

1. Pour l'utilisateur qui démarre la session, son compte IAM utilisateur ou rôle a-t-il été balisé avec `SSMSessionRunAs = os-user-account-name` ?

   Si oui, le nom d'utilisateur existe-t-il sur l'instance ? Si c'est le cas, démarrer la session. Si ce n'est pas le cas, ne pas autoriser une session à démarrer.

   Si le compte ou rôle de l'IAMutilisateur n'a pas été balisé avec `SSMSessionRunAs = os-user-account-name` , passez à l'étape 2.
2. Si le compte ou rôle de l'IAMutilisateur n'a pas été balisé avec `SSMSessionRunAs = os-user-account-name` , un nom d'utilisateur de système d'exploitation a-t-il été spécifié dans les AWS préférences du Session Manager compte ?

   Si oui, le nom d'utilisateur existe-t-il sur l'instance ? Si c'est le cas, démarrer la session. Si ce n'est pas le cas, ne pas autoriser une session à démarrer.

   À ce stade, Session Manager ne rebascule pas sur le compte par défaut `ssm-user`. En d'autres termes, l'activation de la prise en charge de Run As empêche le démarrage des sessions d'être démarrées à l'aide d'un compte `ssm-user` sur une instance.

Pour activer la prise en charge de Run As pour les instances Linux

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Session Manager.
3. Sélectionnez l'onglet Préférences, puis Modifier.
4. Cochez la case à cocher en regard de Enable Run As support for Linux instances (Activation de la prise en charge de Run As pour les instances Linux).
5. Procédez de l'une des façons suivantes :

   - Option 1 : Pour (Optional) Enter an operating system user name for starting sessions ((Facultatif) Saisir un nom d'utilisateur de système d'exploitation pour le démarrage des sessions), entrez le nom du compte utilisateur de système d'exploitation sur l'instance cible que vous souhaitez utiliser pour démarrer les sessions.
   - Option 2 : Choisissez le lien de la Console IAM. Dans le panneau de navigation, sélectionnez Users (Utilisateurs) ou Roles (Rôles). Choisissez l'entité (utilisateur ou rôle) à laquelle ajouter des balises, puis sélectionnez l'onglet Tags (Balises). Entrez `SSMSessionRunAs` pour le nom de la clé. Saisissez le nom d'un compte utilisateur sur votre instance cible pour la valeur de la clé. Sélectionnez Save Changes (Enregistrer les modifications).

     Voici un exemple pour .

6. Choisissez Save.

Activer le chiffrement de clé AWS KMS des données de session (console)

Utilisez AWS Key Management Service (AWS KMS) pour créer et gérer des clés. Avec AWS KMS, vous pouvez contrôler l'utilisation de chiffrement dans un large éventail de services AWS et dans vos applications. Vous pouvez spécifier que les données de session transmises entre vos instances Amazon Elastic Compute Cloud (Amazon EC2) et les machines locales des utilisateurs de votre AWS compte sont chiffrées à l'aide du chiffrement AWS KMS de clé. (En plus du chiffrement TLS 1.2 déjà fourni par défaut par .) Le chiffrement de AWS clé pour les sessions s'effectue à l'aide d'une clé principale client (CMK) qui est créée dans AWS KMS .AWS KMS Pour utiliser l'option permettant de chiffrer les données de session à l'aide d'une clé CMK créée dans AWS KMS , la version 2.3.539.0 ou ultérieure d' Agent SSM doit être installée sur l'instance gérée.

> **Note**
>
> Vous devez activer le chiffrement AWS KMS pour réinitialiser les mots de passe sur vos instances gérées à partir de la console Systems Manager. Pour plus d'informations, consultez la section Réinitialiser un mot de passe sur une Instance gérée (p. 861).

Vous pouvez utiliser une clé que vous avez créée dans votre compte AWS. Vous pouvez également utiliser une clé qui a été créée dans un compte AWS différent. Le créateur de la clé dans un autre AWS compte doit vous fournir les autorisations nécessaires pour utiliser la clé.

Une fois que vous avez activé le chiffrement de clé AWS KMS pour vos données de session, les utilisateurs qui démarrent les sessions et les instances auxquelles ils se connectent doivent avoir l'autorisation d'utiliser la clé. Vous fournissez l'autorisation d'utiliser la clé CMK avec Session Manager via IAM des stratégies. Pour plus d'informations, consultez les rubriques suivantes :

- Ajoutez des autorisations CMK pour les utilisateurs de votre compte : Démarrage rapide - Stratégies IAM par défaut pour Session Manager (p. 885).
- Ajoutez des autorisations CMK pour les instances de votre compte : Étape 2 : Vérifier ou créer un profil d'instance IAM avec des autorisations Session Manager (p. 876).

Pour de plus amples informations sur la création et la gestion des clés AWS KMS, veuillez consulter AWS Key Management Service Developer Guide.

Pour de plus amples informations sur l'utilisation de AWS CLI pour activer le chiffrement de clé AWS KMS des données de session dans votre compte, veuillez consulter Créer des préférences Session Manager (ligne de commande) (p. 903) ou Mettre à jour les préférences Session Manager (ligne de commande) (p. 906).

Note

Des frais sont facturés pour l'utilisation d' CMKs . Pour plus d'informations, consultez AWS Key Management Service la tarification.

Pour activer le chiffrement de clé AWS KMS des données de session (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Session Manager.
3. Sélectionnez l'onglet Préférences, puis Modifier.
4. Activez la case à cocher en regard de Key Management Service (KMS).
5. Procédez de l'une des façons suivantes :

   - Cliquez sur le bouton en regard de Sélectionner une clé AWS KMS dans mon compte actuel, puis sélectionnez une clé dans la liste.

     -ou-

     Choisissez le bouton en regard de Entrer un alias de clé KMS ou un ARN de clé KMS. Saisissez manuellement un alias AWS KMS de clé pour une clé créée dans votre compte actuel, ou saisissez l'Amazon Resource Name (ARN) de clé pour une clé dans un autre compte. Voici quelques exemples :

     - Alias de clé : `alias/my-kms-key-alias`
     - ARN de clé : `arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-12345EXAMPLE`

     -ou-

     Choisissez Créer une nouvelle clé pour créer une nouvelle clé CMK dans votre compte. Une fois que vous avez créé la nouvelle clé, revenez à l'onglet Préférences et sélectionnez la clé pour chiffrer les données de session dans votre compte.

   Pour de plus amples informations sur les clés, veuillez consulter Autoriser les comptes AWS externes à accéder à une clé CMK dans le AWS Key Management Service Developer Guide.
6. Choisissez Save.

Créer des préférences Session Manager (ligne de commande)

La procédure suivante décrit comment utiliser votre outil de ligne de commande préféré pour créer des préférences Session Manager pour votre compte dans la région AWS sélectionnée. Utilisez Session Manager les préférences pour spécifier des options pour la journalisation des données de session dans un compartiment Amazon Simple Storage Service (Amazon S3) ou un groupe Amazon CloudWatch Logs de journaux . Vous pouvez également utiliser des préférences Session Manager pour chiffrer vos données de session.

Pour de plus amples informations sur l'utilisation des outils de ligne de commande afin de mettre à jour les préférences Session Manager existantes, veuillez consulter Mettre à jour les préférences Session Manager (ligne de commande) (p. 906).

Note

Vous pouvez utiliser cette procédure pour créer des documents de session personnalisés pour vos Session Manager préférences qui remplacent les paramètres au niveau du compte. Lorsque vous créez vos documents de session personnalisés, spécifiez une valeur autre que `SSM-SessionManagerRunShell` pour le paramètre de nom et modifiez les entrées selon vos

besoins. Pour utiliser vos documents de session personnalisés, vous devez fournir le nom de votre document de session personnalisé pour le `--document-name` paramètre lorsque vous démarrez une session à partir de l' AWS Command Line Interface (AWS CLI). Lorsque vous démarrez une session à partir de la console, vous ne pouvez pas spécifier de documents de session personnalisés.

Pour créer des préférences Session Manager (ligne de commande)

1. Créez un fichier JSON sur votre ordinateur local avec un nom tel que `SessionManagerRunShell.json` , puis collez le contenu suivant dans celui-ci.

```
{
    "schemaVersion": "1.0",
    "description": "Document to hold regional settings for Session Manager",
    "sessionType": "Standard_Stream",
    "inputs": {
        "s3BucketName": "",
        "s3KeyPrefix": "",
        "s3EncryptionEnabled": true,
        "cloudWatchLogGroupName": "",
        "cloudWatchEncryptionEnabled": true,
        "kmsKeyId": "",
        "runAsEnabled": false,
        "runAsDefaultUser": "",
        "shellProfile": {
            "windows": "date",
            "linux": "pwd;ls"
        }
    }
}
```

2. Spécifiez où vous souhaitez envoyer les données de session. Vous pouvez spécifier un nom de compartiment S3 (avec un préfixe facultatif) ou un nom de groupe de journaux CloudWatch Logs. Si vous souhaitez chiffrer davantage les données entre le client local et les instances Amazon Elastic Compute Cloud (Amazon EC2), fournissez la AWS KMS clé à utiliser pour le chiffrement. Voici un exemple pour .

```
{
  "schemaVersion": "1.0",
  "description": "Document to hold regional settings for Session Manager",
  "sessionType": "Standard_Stream",
  "inputs": {
    "s3BucketName": "DOC-EXAMPLE-BUCKET",
    "s3KeyPrefix": "MyBucketPrefix",
    "s3EncryptionEnabled": true,
    "cloudWatchLogGroupName": "MyLogGroupName",
    "cloudWatchEncryptionEnabled": true,
    "kmsKeyId": "MyKMSKeyID",
    "runAsEnabled": true,
    "runAsDefaultUser": "MyDefaultRunAsUser",
    "shellProfile": {
        "windows": "MyCommands",
        "linux": "MyCommands"
    }
  }
}
```

> **Note**
>
> Si vous ne souhaitez pas chiffrer les données des journaux de session, remplacez par `true` pour `false` .`s3EncryptionEnabled`

Vous pouvez supprimer les lignes pour ces options si vous n'envoyez pas des journaux à un compartiment S3 ou à un groupe de journaux CloudWatch Logs, ou si vous ne souhaitez pas activer la prise en charge de Run As pour les des sessions dans votre compte. Assurez-vous que la dernière ligne de la `inputs` section ne se termine pas par une virgule.
Si vous ajoutez un ID de clé AWS KMS pour chiffrer vos données de session, les utilisateurs qui démarrent les sessions et les instances auxquelles ils se connectent doivent avoir l'autorisation d'utiliser la clé. Vous fournissez l'autorisation d'utiliser la clé CMK avec Session Manager via IAM des stratégies. Pour plus d'informations, consultez les rubriques suivantes :

- Ajoutez des autorisations CMK pour les utilisateurs de votre compte :Démarrage rapide - Stratégies IAM par défaut pour Session Manager (p. 885)
- Ajoutez des autorisations CMK pour les instances de votre compte :Étape 2 : Vérifier ou créer un profil d'instance IAM avec des autorisations Session Manager (p. 876)

3. Enregistrez le fichier .
4. Dans le répertoire où vous avez créé le fichier JSON, exécutez la commande suivante.

Linux

```
aws ssm create-document \
    --name SSM-SessionManagerRunShell \
    --content "file://SessionManagerRunShell.json" \
    --document-type "Session" \
    --document-format JSON
```

Windows

```
aws ssm create-document ^
    --name SSM-SessionManagerRunShell ^
    --content "file://SessionManagerRunShell.json" ^
    --document-type "Session" ^
    --document-format JSON
```

PowerShell

```
New-SSMDocument `
    -Name "SSM-SessionManagerRunShell" `
    -Content (Get-Content -Raw SessionManagerRunShell.json) `
    -DocumentType "Session" `
    -DocumentFormat JSON
```

Si elle aboutit, la commande renvoie un résultat semblable au suivant :

```
{
    "DocumentDescription": {
        "Status": "Creating",
        "Hash": "ce4fd0a2ab9b0fae759004ba603174c3ec2231f21a81db8690a33eb66EXAMPLE",
        "Name": "SSM-SessionManagerRunShell",
        "Tags": [],
        "DocumentType": "Session",
        "PlatformTypes": [
            "Windows",
            "Linux"
        ],
        "DocumentVersion": "1",
        "HashType": "Sha256",
        "CreatedDate": 1547750660.918,
        "Owner": "111122223333",
```

```
        "SchemaVersion": "1.0",
        "DefaultVersion": "1",
        "DocumentFormat": "JSON",
        "LatestVersion": "1"
    }
}
```

Mettre à jour les préférences Session Manager (ligne de commande)

La procédure suivante décrit comment utiliser l'outil de ligne de commande préféré pour modifier les préférences Session Manager de votre compte dans la région AWS sélectionnée. Utilisez Session Manager les préférences pour spécifier des options pour la journalisation des données de session dans un compartiment Amazon Simple Storage Service (Amazon S3) ou un groupe Amazon CloudWatch Logs de journaux . Vous pouvez également utiliser des préférences Session Manager pour chiffrer vos données de session.

Pour mettre à jour les préférences Session Manager (ligne de commande)

1. Créez un fichier JSON sur votre ordinateur local avec un nom tel que `SessionManagerRunShell.json` , puis collez le contenu suivant dans celui-ci.

```
{
    "schemaVersion": "1.0",
    "description": "Document to hold regional settings for Session Manager",
    "sessionType": "Standard_Stream",
    "inputs": {
        "s3BucketName": "",
        "s3KeyPrefix": "",
        "s3EncryptionEnabled": true,
        "cloudWatchLogGroupName": "",
        "cloudWatchEncryptionEnabled": true,
        "kmsKeyId": "",
        "runAsEnabled": true,
        "runAsDefaultUser": "",
        "shellProfile": {
            "windows": "date",
            "linux": "pwd;ls"
        }
    }
}
```

2. Spécifiez où vous souhaitez envoyer les données de session. Vous pouvez spécifier un nom de compartiment S3 (avec un préfixe facultatif) ou un nom de groupe de journaux CloudWatch Logs. Si vous souhaitez chiffrer davantage les données entre le client local et les instances Amazon Elastic Compute Cloud (Amazon EC2), fournissez la clé AWS Key Management Service (AWS KMS) à utiliser pour le chiffrement. Voici un exemple pour .

```
{
  "schemaVersion": "1.0",
  "description": "Document to hold regional settings for Session Manager",
  "sessionType": "Standard_Stream",
  "inputs": {
    "s3BucketName": "DOC-EXAMPLE-BUCKET",
    "s3KeyPrefix": "MyBucketPrefix",
    "s3EncryptionEnabled": true,
    "cloudWatchLogGroupName": "MyLogGroupName",
    "cloudWatchEncryptionEnabled": true,
    "kmsKeyId": "MyKMSKeyID",
    "runAsEnabled": true,
    "runAsDefaultUser": "MyDefaultRunAsUser",
    "shellProfile": {
```

```
        "windows": "MyCommands",
        "linux": "MyCommands"
    }
  }
}
```

Note

Si vous ne souhaitez pas chiffrer les données des journaux de session, remplacez par `true` pour `false` `s3EncryptionEnabled`
Vous pouvez supprimer les lignes pour ces options si vous n'envoyez pas des journaux à un compartiment S3 ou à un groupe de journaux CloudWatch Logs, ou si vous ne souhaitez pas activer la prise en charge de Run As pour les des sessions dans votre compte. Assurez-vous que la dernière ligne de la `inputs` section ne se termine pas par une virgule.
Si vous ajoutez un ID de clé AWS KMS pour chiffrer vos données de session, les utilisateurs qui démarrent les sessions et les instances auxquelles ils se connectent doivent avoir l'autorisation d'utiliser la clé. Vous fournissez l'autorisation d'utiliser la clé CMK avec Session Manager via des stratégies IAM. Pour plus d'informations, consultez les rubriques suivantes :

- Ajoutez des autorisations CMK pour les utilisateurs de votre compte : Démarrage rapide - Stratégies IAM par défaut pour Session Manager (p. 885).
- Ajoutez des autorisations CMK pour les instances de votre compte : Étape 2 : Vérifier ou créer un profil d'instance IAM avec des autorisations Session Manager (p. 876).

3. Enregistrez le fichier .
4. Dans le répertoire où vous avez créé le fichier JSON, exécutez la commande suivante.

Linux

```
aws ssm update-document \
    --name "SSM-SessionManagerRunShell" \
    --content "file://SessionManagerRunShell.json" \
    --document-version "\$LATEST"
```

Windows

```
aws ssm update-document ^
    --name "SSM-SessionManagerRunShell" ^
    --content "file://SessionManagerRunShell.json" ^
    --document-version "$LATEST"
```

PowerShell

```
Update-SSMDocument `
    -Name "SSM-SessionManagerRunShell" `
    -Content (Get-Content -Raw SessionManagerRunShell.json) `
    -DocumentVersion '$LATEST'
```

Si elle aboutit, la commande renvoie un résultat semblable au suivant :

```
{
    "DocumentDescription": {
        "Status": "Updating",
        "Hash": "ce4fd0a2ab9b0fae759004ba603174c3ec2231f21a81db8690a33eb66EXAMPLE",
        "Name": "SSM-SessionManagerRunShell",
        "Tags": [],
        "DocumentType": "Session",
```

```
        "PlatformTypes": [
            "Windows",
            "Linux"
        ],
        "DocumentVersion": "2",
        "HashType": "Sha256",
        "CreatedDate": 1537206341.565,
        "Owner": "111122223333",
        "SchemaVersion": "1.0",
        "DefaultVersion": "1",
        "DocumentFormat": "JSON",
        "LatestVersion": "2"
    }
}
```

Étape 5 : (Facultatif) Restreindre l'accès aux commandes dans une session

Vous pouvez restreindre les commandes qu'un utilisateur peut exécuter dans une session Session Manager en créant un document SSM personnalisé de type `Session`. Dans le contenu du document, définissez la commande qui est exécutée lorsque l'utilisateur démarre une session et les paramètres qu'il peut fournir à la commande. Ces commandes sont également appelées « commandes interactives ». Le `Session` document `schemaVersion` doit être au format 1.0 et celui `sessionType` du document doit être au format `InteractiveCommands` . Vous pouvez ensuite créer des stratégies AWS Identity and Access Management (IAM) qui autorisent les utilisateurs à accéder uniquement aux `Session` documents que vous définissez. Pour de plus amples informations sur l'utilisation des stratégies IAM pour restreindre l'accès aux commandes dans une session, veuillez consulter Exemples de stratégie IAM pour les commandes interactives (p. 910).

L'utilisateur spécifie le document autorisé dans l'option `--document-name` de la commande `start-session` et fournit toutes les valeurs de paramètre nécessaires pour la commande dans l'option `--parameters`. Pour de plus amples informations sur l'exécution des commandes interactives, veuillez consulter Démarrage d'une session (commandes interactives) (p. 927).

La procédure suivante explique comment créer un document SSM personnalisé de type `Session` qui définit la commande qu'un utilisateur est autorisé à exécuter.

Restreindre l'accès aux commandes dans une session (console)

Pour restreindre les commandes qu'un utilisateur peut exécuter dans une session Session Manager (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Documents.
3. Choisissez Create command or session (Créer une commande ou une session).
4. Pour Name (Nom), entrez un nom descriptif pour le document.
5. Pour Document type (Type de document), choisissez Session document (Document de session).
6. Entrez le contenu de votre document qui définit la commande qu'un utilisateur peut exécuter dans une session Session Manager au format JSON ou YAML, comme illustré dans l'exemple suivant.

   YAML

```
---
schemaVersion: '1.0'
description: Document to view a log file on a Linux instance
```

```
sessionType: InteractiveCommands
parameters:
  logpath:
    type: String
    description: The log file path to read.
    default: "/var/log/amazon/ssm/amazon-ssm-agent.log"
    allowedPattern: "^[a-zA-Z0-9-_/]+(.log)$"
properties:
  linux:
    commands: "tail -f {{ logpath }}"
    runAsElevated: true
```

JSON

```
{
    "schemaVersion": "1.0",
    "description": "Document to view a log file on a Linux instance",
    "sessionType": "InteractiveCommands",
    "parameters": {
        "logpath": {
            "type": "String",
            "description": "The log file path to read.",
            "default": "/var/log/amazon/ssm/amazon-ssm-agent.log",
            "allowedPattern": "^[a-zA-Z0-9-_/]+(.log)$"
        }
    },
    "properties": {
        "linux": {
            "commands": "tail -f {{ logpath }}",
            "runAsElevated": true
        }
    }
}
```

7. Choisissez Create document (Créer un document).

Restreindre l'accès aux commandes dans une session (ligne de commande)

Avant de commencer

Si vous ne l'avez pas déjà fait, installez et configurez l' AWS Command Line Interface (AWS CLI) ou l' Outils AWS pour PowerShell . Pour plus d'informations, consultez la section Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

Pour restreindre les commandes qu'un utilisateur peut exécuter dans une session Session Manager (ligne de commande)

1. Créez un fichier JSON ou YAML pour le contenu de votre document qui définit la commande qu'un utilisateur peut exécuter dans une session Session Manager, comme illustré dans l'exemple suivant.

   YAML

```
---
schemaVersion: '1.0'
description: Document to view a log file on a Linux instance
sessionType: InteractiveCommands
parameters:
  logpath:
    type: String
    description: The log file path to read.
    default: "/var/log/amazon/ssm/amazon-ssm-agent.log"
```

```
    allowedPattern: "^[a-zA-Z0-9-_/]+(.log)$"
properties:
  linux:
    commands: "tail -f {{ logpath }}"
    runAsElevated: true
```

JSON

```
{
    "schemaVersion": "1.0",
    "description": "Document to view a log file on a Linux instance",
    "sessionType": "InteractiveCommands",
    "parameters": {
        "logpath": {
            "type": "String",
            "description": "The log file path to read.",
            "default": "/var/log/amazon/ssm/amazon-ssm-agent.log",
            "allowedPattern": "^[a-zA-Z0-9-_/]+(.log)$"
        }
    },
    "properties": {
        "linux": {
            "commands": "tail -f {{ logpath }}",
            "runAsElevated": true
        }
    }
}
```

2. Exécutez les commandes suivantes pour créer un document SSM en utilisant votre contenu qui définit la commande qu'un utilisateur peut exécuter dans une session Session Manager.

   Linux

```
aws ssm create-document \
    --content file://path/to/file/documentContent.json \
    --name "exampleAllowedSessionDocument" \
    --document-type "Session"
```

   Windows

```
aws ssm create-document ^
    --content file://C:\path\to\file\documentContent.json ^
    --name "exampleAllowedSessionDocument" ^
    --document-type "Session"
```

   PowerShell

```
$json = Get-Content -Path "C:\path\to\file\documentContent.json" | Out-String
New-SSMDocument `
    -Content $json `
    -Name "exampleAllowedSessionDocument" `
    -DocumentType "Session"
```

Exemples de stratégie IAM pour les commandes interactives

Vous pouvez créer des stratégies IAM permettant aux utilisateurs d'accéder uniquement aux documents `Session` que vous définissez. Les commandes qu'un utilisateur peut exécuter dans une session Session Manager sont ainsi limitées uniquement aux commandes définies dans vos documents SSM personnalisés de type `Session`.

Autoriser un utilisateur à exécuter une commande interactive sur une seule instance

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":"ssm:StartSession",
         "Resource":[
            "arn:aws:ec2:us-west-2:987654321098:instance/i-02573cafcfEXAMPLE",
            "arn:aws:ssm:us-west-2:987654321098:document/exampleAllowedSessionDocument"
         ],
         "Condition":{
            "BoolIfExists":{
               "ssm:SessionDocumentAccessCheck":"true"
            }
         }
      }
   ]
}
```

Autoriser un utilisateur à exécuter une commande interactive sur toutes les instances

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":"ssm:StartSession",
         "Resource":[
            "arn:aws:ec2:us-west-2:987654321098:instance/*",
            "arn:aws:ssm:us-west-2:987654321098:document/exampleAllowedSessionDocument"
         ],
         "Condition":{
            "BoolIfExists":{
               "ssm:SessionDocumentAccessCheck":"true"
            }
         }
      }
   ]
}
```

Autoriser un utilisateur à exécuter plusieurs commandes interactives sur toutes les instances

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":"ssm:StartSession",
         "Resource":[
            "arn:aws:ec2:us-west-2:987654321098:instance/*",
            "arn:aws:ssm:us-
west-2:987654321098:document/exampleAllowedSessionDocument",
            "arn:aws:ssm:us-
west-2:987654321098:document/exampleAllowedSessionDocument2"
         ],
         "Condition":{
            "BoolIfExists":{
               "ssm:SessionDocumentAccessCheck":"true"
            }
         }
      }
```

```
    ]
}
```

Étape 6 : (Facultatif) Utilisez AWS PrivateLink pour configurer un point de terminaison de VPC pourSession Manager

Vous pouvez améliorer davantage la sécurité de vos instances gérées en configurant AWS Systems Manager pour qu'il utilise un point de terminaison d'interface Virtual Private Cloud (VPC). Les points de terminaison d'interface reposent sur AWS PrivateLink , une technologie qui vous permet d'accéder de façon privée à Amazon EC2 et à l'aide d'adresses IP privées.Systems ManagerAPIs

limite AWS PrivateLink l'ensemble du trafic réseau entre vos instances gérées, et Systems Manager au réseau Amazon.Amazon EC2 (Les instances gérées n'ont pas accès à Internet.) En outre, vous n'avez pas besoin d'une passerelle Internet, d'un périphérique NAT ou d'une passerelle réseau privé virtuel.

Pour plus d'informations sur la création d'un VPC, consultez (Facultatif) Créer un point de terminaison de VPC (p. 42) .

L'alternative à l'utilisation d'un point de terminaison de VPC est l'activation de l'accès Internet sortant sur vos instances gérées. Dans ce cas, les instances gérées doivent également autoriser le trafic sortant HTTPS (port 443) vers les points de terminaison suivants :

- `ec2messages.`*`region`*`.amazonaws.com`
- `ssm.`*`region`*`.amazonaws.com`
- `ssmmessages.`*`region`*`.amazonaws.com`

Systems Managerutilise le dernier de ces points de terminaison, `ssmmessages.`*`region`*`.amazonaws.com` , pour effectuer des appels depuis Agent SSM vers le Session Manager service dans le cloud.

Pour plus d'informations sur les points de terminaison requis pour Systems Manager , consultez Référence : ec2messages, ssmmessages et autres appels d'API (p. 1399) .

Étape 7 : (Facultatif) Désactivez ou activez les autorisations administratives de compte ssm-user

À partir de la version 2.3.50.0 de l'Agent SSM, l'agent crée un compte utilisateur local appelé `ssm-user` et l'ajoute à l'utilisateur `/etc/sudoers` (Linux) ou au groupe Administrateurs (Windows) chaque fois que l'agent démarre. Sur les versions de l'agent antérieures à la version 2.3.612.0, le compte est créé la première fois qu'Agent SSMil démarre ou redémarre après l'installation. Sur la version 2.3.612.0 et version ultérieure, le compte `ssm-user` est créé la première fois qu'une session est démarrée sur une instance. Il s'`ssm-user`agit de l'utilisateur du système d'exploitation (OS) par défaut lorsqu'une Session Manager session est démarrée.

Si vous souhaitez interdire aux utilisateurs Session Manager d'exécuter des commandes administratives sur une instance, vous pouvez mettre à jour les autorisations du compte `ssm-user`. Vous pouvez restaurer ces autorisations après qu'elles ont été supprimées.

Rubriques

- Gérer les autorisations de compte sudo ssm-user sous Linux (p. 913)
- Gestion des autorisations administrateur de compte ssm-user sur Windows Server (p. 913)

Gérer les autorisations de compte sudo ssm-user sous Linux

Utilisez l'une des procédures suivantes pour désactiver ou activer les autorisations sudo de compte ssm-user sur les instances Linux.

Utiliser Fonctionnalité Exécuter la commande pour modifier les autorisations sudo ssm-user (console)

- Exécutez la procédure décrite sur la page Exécution des commandes depuis la console (p. 946) en appliquant les valeurs suivantes :
  - Pour Command document (Document de commande), choisissez `AWS-RunShellScript`.
  - Pour supprimer l'accès sudo, dans la zone Paramètres de commande, collez ce qui suit dans la zone Commandes.

    ```
    cd /etc/sudoers.d
    echo "#User rules for ssm-user" > ssm-agent-users
    ```

    -ou-

    Pour restaurer l'accès sudo, dans la zone Command parameters (Paramètres de commande), collez ce qui suit dans la zone Commands (Commandes).

    ```
    cd /etc/sudoers.d
    echo "ssm-user ALL=(ALL) NOPASSWD:ALL" > ssm-agent-users
    ```

Utiliser la ligne de commande pour modifier des autorisations sudo ssm-user (AWS CLI)

1. Connectez-vous à l'instance et exécutez la commande suivante.

   ```
   sudo -s
   ```

2. Modifiez le répertoire de travail à l'aide de la commande suivante.

   ```
   cd /etc/sudoers.d
   ```

3. Ouvrez le fichier nommé `ssm-agent-users` pour le modifier.
4. Pour supprimer l'accès sudo, supprimez la ligne suivante.

   ```
   ssm-user ALL=(ALL) NOPASSWD:ALL
   ```

   -ou-

   Pour restaurer l'accès sudo, ajoutez la ligne suivante.

   ```
   ssm-user ALL=(ALL) NOPASSWD:ALL
   ```

5. Enregistrez le fichier .

Gestion des autorisations administrateur de compte ssm-user sur Windows Server

Utilisez l'une des procédures suivantes pour désactiver ou activer les autorisations administrateur de compte ssm-user sur Windows Server les instances .

Utiliser Fonctionnalité Exécuter la commande pour modifier les autorisations administrateur (console)

- Exécutez la procédure décrite sur la page Exécution des commandes depuis la console (p. 946) en appliquant les valeurs suivantes :

  Pour Command document (Document de commande), choisissez `AWS-RunPowerShellScript`.

  Pour supprimer un accès administratif, dans la zone Command parameters (Paramètres de la commande), collez ce qui suit dans la zone Commands (Commandes).

  ```
  net localgroup "Administrators" "ssm-user" /delete
  ```

  -ou-

  Pour restaurer un accès administratif, dans la zone Command parameters (Paramètres de la commande), collez ce qui suit dans la zone Commands (Commandes).

  ```
  net localgroup "Administrators" "ssm-user" /add
  ```

Utiliser la fenêtre PowerShell d'invite de commande ou pour modifier les autorisations administrateur

1. Connectez-vous à l'instance et ouvrez la fenêtre PowerShell d'invite de commande ou .
2. Pour supprimer un accès administratif, exécutez la commande suivante.

   ```
   net localgroup "Administrators" "ssm-user" /delete
   ```

   -ou-

   Pour restaurer un accès administratif, exécutez la commande suivante.

   ```
   net localgroup "Administrators" "ssm-user" /add
   ```

Utiliser la console Windows pour modifier les autorisations administrateur

1. Connectez-vous à l'instance et ouvrez la fenêtre PowerShell d'invite de commande ou .
2. À partir de la ligne de commande, exécutez `lusrmgr.msc` pour ouvrir la console Utilisateurs et groupes locaux.
3. Ouvrez le répertoire Utilisateurs, puis ssm-user.
4. Dans l'onglet Membre de, effectuez l'une des opérations suivantes :

   - Pour supprimer un accès administrateur, sélectionnez Administrateurs, puis Supprimer.

     -ou-

     Pour restaurer un accès administratif, entrez **`Administrators`** dans la zone de texte, puis choisissez Ajouter.
5. Choisissez OK.

Étape 8 : (Facultatif) Activez les connexions SSH via Session Manager

Vous pouvez permettre aux utilisateurs de votre AWS compte d'utiliser l' AWS Command Line Interface (AWS CLI) pour établir des connexions Secure Shell (SSH) aux instances à l'aide d' Session Manager . Les utilisateurs qui se connectent à l'aide de SSH peuvent également copier des fichiers entre leurs ordinateurs locaux et les instances gérées à l'aide de SCP (Secure Copy Protocol). Vous pouvez utiliser cette fonctionnalité pour vous connecter aux instances sans ouvrir des ports entrants ou gérer des hôtes bastion. Vous pouvez également choisir de désactiver explicitement les connexions SSH avec vos instances via Session Manager.

> **Note**
>
> La journalisation et l'audit ne sont pas disponibles pour les sessions Session Manager qui se connectent via le transfert de port ou SSH. Cela est dû au fait que SSH crypte toutes les données de session et que Session Manager sert uniquement de tunnel pour les connexions SSH.

Pour activer les connexions SSH via Session Manager

1. Sur l'instance gérée vers lequel vous souhaitez activer les connexions SSH, procédez comme suit :

   - Assurez-vous que SSH s'exécute sur l'instance. (Vous pouvez fermer les ports entrants sur l'instance.)
   - Assurez-vous que Agent SSM 2.3.672.0 ou une version ultérieure est installé sur l'instance.

     Pour plus d'informations sur l'installation de Agent SSM sur une instance, consultez les rubriques suivantes :

     - Installation et configuration Agent SSM de sur les instances EC2 pourWindows Server (p. 100).
     - Installation et configuration de l'Agent SSM sur les instances EC2 pour Linux (p. 75)
     - Installer Agent SSM pour un environnement hybride (Windows) (p. 62)
     - Installer Agent SSM pour un environnement hybride (Linux) (p. 58)

       > **Note**
       >
       > Pour utiliser Session Manager avec des serveurs et des machines virtuelles sur site que vous avez activés en tant qu'instances gérées, vous devez utiliser le niveau Advanced-Instances (Instances avancées). Pour de plus amples informations, sur les instances avancées, veuillez consulter Activation du niveau d'instances avancées (p. 853).
2. Sur l'ordinateur local à partir duquel vous souhaitez vous connecter à une instance gérée à l'aide de SSH, procédez comme suit :

   - Assurez-vous que la version 1.1.23.0 ou une version ultérieure du plug-in Session Manager est installée.

     Pour plus d'informations sur l'installation du plug-in Session Manager, consultez (Facultatif) Installer le Session Manager plugin pour l'AWS CLI (p. 917).
   - Mettez à jour le fichier de configuration SSH pour activer l'exécution d'une commande de proxy qui démarre une session Session Manager et transfère toutes les données via la connexion.

     Linux

     > **Tip**
     >
     > Le fichier de configuration SSH est généralement situé dans `~/.ssh/config`.

     Ajoutez les éléments suivants au fichier de configuration sur l'ordinateur local.

```
# SSH over Session Manager
```

```
host i-* mi-*
    ProxyCommand sh -c "aws ssm start-session --target %h --document-name AWS-
StartSSHSession --parameters 'portNumber=%p'"
```

Windows

Tip

Le fichier de configuration SSH est généralement situé dans `C:\Users\username\.ssh\config`.

Ajoutez les éléments suivants au fichier de configuration sur l'ordinateur local.

```
# SSH over Session Manager
host i-* mi-*
    ProxyCommand C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe "aws
 ssm start-session --target %h --document-name AWS-StartSSHSession --parameters
 portNumber=%p"
```

- Créez ou vérifiez que vous disposez d'un certificat PEM (Privacy Enhanced Mail) ou au minimum d'une clé publique, à utiliser lors de l'établissement de connexions aux instances gérées. Il doit s'agir d'une clé déjà associée à l'instance. Par exemple, pour une instance Amazon Elastic Compute Cloud (Amazon EC2), le fichier de paire de clés que vous avez créé ou sélectionné lors de la création de l'instance. (Vous spécifiez le chemin d'accès au certificat ou à la clé dans la commande de démarrage d'une session. Pour plus d'informations sur le démarrage d'une session à l'aide de SSH, consultez Démarrage d'une session (SSH) (p. 926) .)

Pour activer les connexions SSH via Session Manager

- Option 1 : Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/. Dans le panneau de navigation, choisissez Policies (Stratégies), puis mettez à jour la stratégie d'autorisations permettant à l'utilisateur ou au rôle de votre choix de démarrer les connexions SSH via Session Manager. Par exemple, préparez-vous à modifier la stratégie de démarrage rapide de l'utilisateur que vous avez créée dans Démarrage rapide - Stratégies d'utilisateur final pour Session Manager (p. 885). Ajoutez l'élément suivant à la stratégie.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ssm:StartSession",
            "Resource": [
                "arn:aws:ec2:*:*:instance/instance-id",
                "arn:aws:ssm:*:*:document/AWS-StartSSHSession"
            ]
        }
    ]
}
```

Option 2 : Attachez une stratégie en ligne à une stratégie d'utilisateur à l'aide AWS Management Console de l' , de l' ou de lAWS CLI'API .AWS

A l'aide de la méthode de votre choix, attachez la déclaration de stratégie de l'option 1 à la stratégie dAWS'un utilisateur, d'un groupe ou d'un rôle .

Pour de plus amples informations, veuillez consulter Ajout et suppression d'autorisations d'identité IAM dans le IAM Guide de l'utilisateur.

Pour désactiver les connexions SSH via Session Manager

- Option 1 : Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/. Dans le panneau de navigation, choisissez Policies (Stratégies), puis mettez à jour la stratégie d'autorisations pour l'utilisateur ou le rôle afin de bloquer le démarrage des sessions Session Manager. Par exemple, préparez-vous à modifier la stratégie de démarrage rapide de l'utilisateur que vous avez créée dans Démarrage rapide - Stratégies d'utilisateur final pour Session Manager (p. 885). Ajoutez l'élément suivant à la stratégie ou remplacez toutes les autorisations qui permettent à un utilisateur de démarrer une session.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor1",
            "Effect": "Deny",
            "Action": "ssm:StartSession",
            "Resource": "arn:aws:ssm:*:*:document/AWS-StartSSHSession"
        }
    ]
}
```

  Option 2 : Attachez une stratégie en ligne à une stratégie d'utilisateur à l'aide AWS Management Console de l' , de l' ou de lAWS CLI'API .AWS

  A l'aide de la méthode de votre choix, attachez la déclaration de stratégie de l'option 1 à la stratégie dAWS'un utilisateur, d'un groupe ou d'un rôle .

  Pour de plus amples informations, veuillez consulter Ajout et suppression d'autorisations d'identité IAM dans le IAM Guide de l'utilisateur.

Utilisation des Session Manager

Vous pouvez utiliser la AWS Systems Manager console , la console Amazon Elastic Compute Cloud (Amazon EC2) ou l' AWS Command Line Interface (AWS CLI) pour démarrer des sessions qui vous connectent aux instances EC2 auxquelles votre administrateur système vous a accordé l'accès à l'aide de stratégies AWS Identity and Access Management (IAM). Selon vos autorisations, vous pouvez également afficher des informations sur les sessions, reprendre des sessions inactives qui n'ont pas expiré, et mettre fin à des sessions.

Pour de plus amples informations sur les sessions, veuillez consulter la page Qu'est-ce qu'une session ? (p. 872).

Rubriques

- (Facultatif) Installer le Session Manager plugin pour l'AWS CLI (p. 917)
- Démarrer une session (p. 924)
- Mettre fin à une session (p. 928)
- Afficher l'historique de session (p. 928)

(Facultatif) Installer le Session Manager plugin pour l'AWS CLI

Si vous souhaitez utiliser l' AWS Command Line Interface (AWS CLI) pour démarrer et mettre fin aux sessions qui vous connectent à vos instances gérées, vous devez d'abord installer l' Session Manager plugin sur votre ordinateur local. Le plug-in peut être installé sur les versions prises en charge de Microsoft Windows, macOS , Linux et Ubuntu Server .

Utiliser la dernière version de l'Session Manager plugin

est mis à jour occasionnellement avec des fonctionnalités améliorées.Session Manager plugin Nous vous recommandons de vous assurer régulièrement que vous utilisez la dernière version du plug-in. Pour plus d'informations, consultez la section Dernière version et historique des versions Session Manager plugin (p. 923).

Conditions d'installation préalables

L'AWS CLI version 1.16.12 ou ultérieure doit être installée sur votre ordinateur local pour pouvoir utiliser le Session Manager plugin.

Rubriques

- Installation de l'Session Manager plugin sur Windows (p. 918)
- Installation et désinstallation de l' Session Manager plugin surmacOS (p. 919)
- Installer le Session Manager plugin sur Linux (p. 920)
- Installation de l'Session Manager plugin sur Ubuntu Server (p. 920)
- Vérifier l'installation Session Manager plugin (p. 921)
- (Facultatif) Activer Session Manager plugin la journalisation (p. 921)
- Dernière version et historique des versions Session Manager plugin (p. 923)

Installation de l'Session Manager plugin sur Windows

Vous pouvez installer le Session Manager plugin sur Microsoft Windows Vista ou une version ultérieure à l'aide du programme d'installation autonome.

Lorsque des mises à jour sont publiées, vous devez relancer le processus d'installation pour obtenir la version la plus récente de l'Session Manager plugin.

> **Note**
>
> Pour obtenir de meilleurs résultats, nous vous recommandons de démarrer les sessions sur les clients Windows à l'aide de l'PowerShellapplication Windows version 5 ou ultérieure. Sous Microsoft Windows 10, l'invite de commande offre également une prise en charge fiable des opérations Session Manager.

Pour installer le Session Manager plugin à l'aide du programme d'installation EXE

1. Téléchargez le programme d'installation à l'URL suivante.

   ```
   https://s3.amazonaws.com/session-manager-downloads/plugin/latest/windows/
   SessionManagerPluginSetup.exe
   ```

2. Exécutez le programme d'installation téléchargé et suivez les instructions à l'écran.

   Laissez la zone de l'emplacement d'installation vide pour installer le plug-in dans le répertoire par défaut.

   - `C:\%PROGRAMFILES%\Amazon\SessionManagerPlugin\bin\`

3. Vérifiez que l'installation a réussi. Pour plus d'informations, consultez la section Vérifier l'installation Session Manager plugin (p. 921).

   > **Note**
   >
   > Si Windows ne parvient pas à localiser le fichier exécutable, vous devrez peut-être rouvrir l'invite de commande ou ajouter manuellement le répertoire d'installation à votre variable

d'environnement `PATH`. Pour en savoir plus, consultez la rubrique de dépannage Session Manager plugin n'est pas ajouté automatiquement au chemin de la ligne de commande (Windows) (p. 940).

Installation et désinstallation de l' Session Manager plugin surmacOS

Vous pouvez installer l' Session Manager plugin sur macOS à l'aide du programme d'installation fourni.

> **Important**
>
> Le programme d'installation fourni ne prend pas en charge l'installation dans des chemins contenant des espaces.

Pour installer le Session Manager plugin à l'aide du programme d'installation fourni (macOS)

1. Téléchargez le programme d'installation fourni.

```
curl "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/mac/
sessionmanager-bundle.zip" -o "sessionmanager-bundle.zip"
```

2. Décompressez le package.

```
unzip sessionmanager-bundle.zip
```

3. Exécutez la commande d'installation.

```
sudo ./sessionmanager-bundle/install -i /usr/local/sessionmanagerplugin -b /usr/local/
bin/session-manager-plugin
```

> **Note**
>
> Le plug-in nécessite les versions Python 2.6.5 ou ultérieure, ou Python 3.3. Par défaut, le script d'installation s'exécute sous la version système par défaut de Python. Si vous avez installé une autre version de Python et souhaitez l'utiliser pour installer le Session Manager plugin, exécutez le script d'installation avec cette version dans le chemin d'accès absolu au fichier exécutable Python. Voici un exemple pour .
>
> ```
> sudo /usr/local/bin/python3.6 sessionmanager-bundle/install -i /usr/local/
> sessionmanagerplugin -b /usr/local/bin/session-manager-plugin
> ```

Le programme d'installation installe le Session Manager plugin sous `/usr/local/sessionmanagerplugin` et crée le lien symbolique `session-manager-plugin` dans le répertoire `/usr/local/bin`. Cela permet d'éviter de devoir spécifier le répertoire d'installation dans la variable utilisateur `$PATH`.

Pour voir une explication de `-i` et `-b` , utilisez le `-h` Option .

```
./sessionmanager-bundle/install -h
```

4. Vérifiez que l'installation a réussi. Pour plus d'informations, consultez la section Vérifier l'installation Session Manager plugin (p. 921).

> **Note**
>
> Si vous souhaitez désinstaller le plug-in, exécutez les deux commandes suivantes dans l'ordre indiqué.

```
sudo rm -rf /usr/local/sessionmanagerplugin
```

```
sudo rm /usr/local/bin/session-manager-plugin
```

Installer le Session Manager plugin sur Linux

1. Téléchargez le package Session Manager plugin RPM.

   - Intel 64-bit (x86_64)

     ```
     curl "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/linux_64bit/
     session-manager-plugin.rpm" -o "session-manager-plugin.rpm"
     ```

   - Intel 32-bit (x86)

     ```
     curl "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/linux_32bit/
     session-manager-plugin.rpm" -o "session-manager-plugin.rpm"
     ```

   - ARM 64-bit (arm64)

     ```
     curl "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/linux_arm64/
     session-manager-plugin.rpm" -o "session-manager-plugin.rpm"
     ```

2. Exécutez la commande d'installation.

   ```
   sudo yum install -y session-manager-plugin.rpm
   ```

3. Vérifiez que l'installation a réussi. Pour plus d'informations, consultez la section Vérifier l'installation Session Manager plugin (p. 921).

   Note

   Si vous souhaitez désinstaller le plug-in, exécutez `sudo yum erase session-manager-plugin -y`

Installation de l'Session Manager plugin sur Ubuntu Server

1. Téléchargez le package Session Manager plugin deb.

   - Intel 64-bit (x86_64)

     ```
     curl "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/ubuntu_64bit/
     session-manager-plugin.deb" -o "session-manager-plugin.deb"
     ```

   - Intel 32-bit (x86)

     ```
     curl "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/ubuntu_32bit/
     session-manager-plugin.deb" -o "session-manager-plugin.deb"
     ```

   - ARM 64-bit (arm64)

     ```
     curl "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/ubuntu_arm64/
     session-manager-plugin.deb" -o "session-manager-plugin.deb"
     ```

2. Exécutez la commande d'installation.

```
sudo dpkg -i session-manager-plugin.deb
```

3. Vérifiez que l'installation a réussi. Pour plus d'informations, consultez la section Vérifier l'installation Session Manager plugin (p. 921).

   Note

   Si vous souhaitez désinstaller le plug-in, exécutez `sudo dpkg -r session-manager-plugin`

Vérifier l'installation Session Manager plugin

Exécutez les commandes suivantes pour vérifier que l' Session Manager plugin est installée correctement.

```
session-manager-plugin
```

Si l'installation a réussi, le message suivant est renvoyé.

```
The Session Manager plugin is installed successfully. Use the AWS CLI to start a session.
```

Vous pouvez également tester l'installation en exécutant la commande suivante dans l' AWS CLI .

Note

Cette commande fonctionne uniquement si votre administrateur Session Manager vous a accordé les autorisations IAM nécessaires pour accéder à l'instance cible via Session Manager.

```
aws ssm start-session --target id-of-an-instance-you-have-permissions-to-access
```

(Facultatif) Activer Session Manager plugin la journalisation

Le Session Manager plugin inclut une option permettant d'activer la journalisation pour les sessions que vous exécutez. Par défaut, la journalisation est désactivée.

Si vous activez la journalisation, le Session Manager plugin crée des fichiers journaux relatifs à l'activité de l'application (`session-manager-plugin.log`) et aux erreurs (`errors.log`) sur votre ordinateur local.

Rubriques

- Activer la journalisation dans le Session Manager plugin (Windows) (p. 921)
- Activer la journalisation pour l' Session Manager plugin (Linux et macOS) (p. 922)

Activer la journalisation dans le Session Manager plugin (Windows)

1. Recherchez le fichier `seelog.xml.template` du plug-in.

   L'emplacement par défaut est `C:\Program Files\Amazon\SessionManagerPlugin\seelog.xml.template`.
2. Remplacez le nom du fichier par `seelog.xml`.
3. Ouvrez le fichier et remplacez `minlevel="off"` par `minlevel="info"` ou `minlevel="debug"`.

   Note

   Par défaut, les entrées de journal sur l'ouverture d'un canal de données et les sessions de reconnexion sont enregistrées au niveau INFO. Les entrées de flux de données (paquets et accusé de réception) sont enregistrées au niveau DEBUG.

4. Modifiez les autres options de configuration que vous souhaitez modifier. Vous pouvez modifier les options suivantes :

   - Niveau de débogage : Vous pouvez remplacer le niveau de débogage par `formatid="fmtinfo" .outputs formatid="fmtdebug"`
   - Options des fichiers journaux : Vous pouvez apporter des modifications aux options du fichier journal, y compris à l'emplacement où les journaux sont stockés, à l'exception des noms des fichiers journaux.

     Important

     Ne modifiez pas les noms de fichier, la journalisation ne fonctionnerait pas correctement.

```
<rollingfile type="size" filename="C:\%PROGRAMDATA%\Amazon\SessionManagerPlugin\Logs
\session-manager-plugin.log" maxsize="30000000" maxrolls="5"/>
<filter levels="error,critical" formatid="fmterror">
<rollingfile type="size" filename="C:\%PROGRAMDATA%\Amazon\SessionManagerPlugin\Logs
\errors.log" maxsize="10000000" maxrolls="5"/>
```

5. Enregistrez le fichier .

Activer la journalisation pour l' Session Manager plugin (Linux et macOS)

1. Recherchez le fichier `seelog.xml.template` du plug-in.

   L'emplacement par défaut est `/usr/local/sessionmanagerplugin/seelog.xml.template`.
2. Remplacez le nom du fichier par `seelog.xml`.
3. Ouvrez le fichier et remplacez `minlevel="off"` par `minlevel="info"` ou `minlevel="debug"`.

   Note

   Par défaut, les entrées de journal sur l'ouverture des canaux de données et les sessions de reconnexion sont enregistrées au niveau INFO. Les entrées de flux de données (paquets et accusé de réception) sont enregistrées au niveau DEBUG.
4. Modifiez les autres options de configuration que vous souhaitez modifier. Vous pouvez modifier les options suivantes :

   - Niveau de débogage : Vous pouvez modifier le niveau de débogage de `formatid="fmtinfo" àoutputs formatid="fmtdebug"`
   - Options des fichiers journaux : Vous pouvez apporter des modifications aux options du fichier journal, y compris à l'emplacement où les journaux sont stockés, à l'exception des noms des fichiers journaux.

     Important

     Ne modifiez pas les noms de fichier, la journalisation ne fonctionnerait pas correctement.

```
<rollingfile type="size" filename="/usr/local/sessionmanagerplugin/logs/session-
manager-plugin.log" maxsize="30000000" maxrolls="5"/>
<filter levels="error,critical" formatid="fmterror">
<rollingfile type="size" filename="/usr/local/sessionmanagerplugin/logs/errors.log"
 maxsize="10000000" maxrolls="5"/>
```

     Important

     Si vous utilisez le répertoire par défaut spécifié pour le stockage des journaux, vous devez exécuter des commandes de session à l'aide de la commande sudo ou accorder au répertoire dans lequel le plug-in est installé, des autorisations complètes de lecture

et d'écriture. Pour contourner ces restrictions, modifiez l'emplacement de stockage des journaux.

5. Enregistrez le fichier .

Dernière version et historique des versions Session Manager plugin

Votre ordinateur local doit exécuter une version prise en charge du Session Manager plugin. La version minimale actuellement prise en charge est 1.1.17.0. Si vous exécutez une version antérieure, vos opérations Session Manager peuvent échouer.

Pour voir si vous disposez de la dernière version, exécutez la commande suivante dans l' AWS CLI .

> **Note**
>
> La commande renvoie un résultat uniquement si le plug-in est situé dans le répertoire d'installation par défaut de votre type de système d'exploitation. Vous pouvez également vérifier la version dans le contenu du fichier `VERSION`, dans le répertoire où vous avez installé le plug-in.

```
session-manager-plugin --version
```

Le tableau suivant répertorie toutes les versions du Session Manager plugin, ainsi que les fonctionnalités et les améliorations incluses dans chaque version.

Version	Date de publication	Details (Détails)
1.2.7.0	15 octobre 2020	Amélioration : (Sessions de réacheminement de port uniquement) Réduction de la latence et amélioration des performances globales.
1.1.61.0	17 avril 2020	Amélioration : Ajout de la prise en charge d'ARM pour Linux et Ubuntu Server .
1.1.54.0	6 janvier 2020	Correctif de bogue : Gère le scénario de condition de concurrence des paquets abandonnés lorsque l' n'Session Manager pluginest pas prêt.
1.1.50.0	19 novembre 2019	Amélioration : Ajout de la prise en charge du réacheminement d'un port vers un socket unix local.
1.1.35.0	7 novembre 2019	Amélioration : (Sessions de réacheminement de port uniquement) Envoyer une TerminateSession commande à Agent SSM lorsque l'utilisateur local appuie sur Ctrl+C.
1.1.33.0	26 septembre 2019	Amélioration : (Sessions de réacheminement de port uniquement) Envoyez un signal de déconnexion au serveur lorsque le client abandonne la connexion TCP.
1.1.31.0	6 septembre 2019	Amélioration : Mise à jour permettant de conserver la session de réacheminement de port ouverte jusqu'à ce que le serveur distant ferme la connexion.
1.1.26.0	30 juillet 2019	Amélioration : Mise à jour pour limiter le taux de transfert de données au cours d'une session.
1.1.23.0	9 juillet 2019	Amélioration : Ajout de la prise en charge de l'exécution de sessions SSH à l'aide d' Session Manager .

Version	Date de publication	Details (Détails)
1.1.17.0	4 avril 2019	Amélioration : Ajout de la prise en charge du chiffrement des données de session à l'aide d' AWS Key Management Service (AWS KMS).
1.0.37.0	20 septembre 2018	Amélioration : Correctif de bogue pour la version Windows.
1.0.0.0	11 septembre 2018	Version initiale du Session Manager plugin.

Démarrer une session

Vous pouvez utiliser la AWS Systems Manager console , l' Amazon Elastic Compute Cloud (Amazon EC2), l' AWS Command Line Interface (AWS CLI) ou SSH pour démarrer une session.

Rubriques

- Démarrage d'une session (console Systems Manager) (p. 924)
- Démarrage d'une session (console Amazon EC2) (p. 925)
- Démarrage d'une session (AWS CLI) (p. 925)
- Démarrage d'une session (SSH) (p. 926)
- Démarrage d'une session (réacheminement de port) (p. 926)
- Démarrage d'une session (commandes interactives) (p. 927)

Démarrage d'une session (console Systems Manager)

Vous pouvez utiliser la console AWS Systems Manager pour démarrer une session avec une instance dans votre compte.

> **Note**
>
> Avant de démarrer une session, assurez-vous d'avoir effectué les étapes de configuration pour Session Manager. Pour plus d'informations, consultez Configuration d'Session Manager (p. 873).

Pour démarrer une session (console Systems Manager)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le volet de navigation, choisissez Session Manager.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le volet de navigation, puis choisissez Session Manager dans le volet de navigation.
3. Sélectionnez Start session (Démarrer une session).
4. Dans la liste Target instances (Instances cibles), sélectionnez la case d'option située à gauche de l'instance à laquelle vous souhaitez vous connecter.

   Si une instance à laquelle vous souhaitez vous connecter ne figure pas dans la liste, ou figure dans la liste des rapports avec le message d'erreur « L'instance que vous avez sélectionnée n'est pas configurée pour utiliser Session Manager », consultez la page L'instance n'est pas disponible ou n'est pas configurée pour Session Manager (p. 939) pour suivre la procédure de dépannage.
5. Sélectionnez Start session (Démarrer une session).

Une fois la connexion effectuée, vous pouvez exécuter des commandes bash (Linux) ou PowerShell des commandes (Windows) comme vous le feriez via n'importe quel autre type de connexion.

Démarrage d'une session (console Amazon EC2)

Vous pouvez utiliser la console Amazon Elastic Compute Cloud (Amazon EC2) pour démarrer une session avec une instance dans votre compte.

> **Note**
>
> Si vous recevez un message d'erreur indiquant que vous n'êtes pas autorisé à exécuter une ou plusieurs actions Systems Manager (`ssm:command-name`), vous devez contacter votre administrateur pour obtenir de l'aide. Votre administrateur est la personne qui vous a fourni votre nom d'utilisateur et votre mot de passe. Demandez à cette personne de mettre à jour vos stratégies afin de vous autoriser à démarrer des sessions à partir de la console Amazon EC2. Si vous êtes administrateur, consultez Démarrage rapide - Stratégies IAM par défaut pour Session Manager (p. 885) pour plus d'informations.

Pour démarrer une session (console Amazon EC2)

1. Ouvrez la console Amazon EC2 à l'adresse https://console.aws.amazon.com/ec2/.
2. Dans le panneau de navigation, choisissez Instances.
3. Sélectionnez l'instance, puis choisissez Connect (Connexion).
4. Pour Méthode de connexion, choisissez Session Manager.
5. Choisissez Connect (Se connecter).

Une fois la connexion effectuée, vous pouvez exécuter des commandes bash (Linux) ou PowerShell des commandes (Windows) comme vous le feriez via n'importe quel autre type de connexion.

Démarrage d'une session (AWS CLI)

Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

Pour démarrer une session à l'aide AWS CLI de l' , exécutez la commande suivante.

> **Note**
>
> Avant de démarrer une session, assurez-vous d'avoir effectué les étapes de configuration pour Session Manager. Pour plus d'informations, consultez Configuration d'Session Manager (p. 873). Pour que vous puissiez utiliser l'AWS CLI pour exécuter des commandes de session, le Session Manager plugin doit également être installé sur votre ordinateur local. Pour de plus amples informations, veuillez consulter (Facultatif) Installer le Session Manager plugin pour l'AWS CLI (p. 917).

```
aws ssm start-session \
    --target instance-id
```

`instance-id` représente l'ID d'une instance configurée pour être utilisée avec AWS Systems Manager et sa Session Manager capacité, par exemple `i-02573cafcfEXAMPLE` .

Pour de plus amples informations sur les autres options possibles avec la commande start-session, veuillez consulter start-session dans le document AWS Systems Manager section of the AWS CLI Command Reference.

Démarrage d'une session (SSH)

Pour démarrer une session Session Manager SSH, la version 2.3.672.0 ou ultérieure d' Agent SSM doit être installée sur l'instance gérée.

Configuration requise pour la connexion SSH

Prenez note des exigences et limitations suivantes pour les connexions de session à l'aide de SSH :

- Votre instance cible doit être configurée pour prendre en charge les connexions SSH. Pour plus d'informations, consultez la section (Facultatif) Activez les connexions SSH via Session Manager (p. 915).
- Vous devez utiliser l'utilisateur sur l'instance associée au certificat PEM (Privacy Enhanced Mail), et non sur le compte `ssm-user` utilisé pour d'autres types de connexions de session. Par exemple, sur les instances EC2 pour Linux, l'utilisateur par défaut est `ec2-user` . Pour plus d'informations sur l'identification de l'utilisateur par défaut pour chaque type d'instance, consultez Obtenir des informations sur votre instance dans le Amazon EC2 Guide de l'utilisateur pour les instances Linux .
- La journalisation et l'audit ne sont pas disponibles pour les sessions Session Manager qui se connectent via le transfert de port ou SSH. Cela est dû au fait que SSH crypte toutes les données de session et que Session Manager sert uniquement de tunnel pour les connexions SSH.

> **Note**
>
> Avant de démarrer une session, assurez-vous d'avoir effectué les étapes de configuration pour Session Manager. Pour plus d'informations, consultez Configuration d'Session Manager (p. 873).

Pour démarrer une session à l'aide de SSH, exécutez la commande suivante.

```
ssh -i /path/my-key-pair.pem username@instance-id
```

`/path/my-key-pair.pem` représente le chemin d'accès au certificat PEM associé à l'instance. Par exemple, pour une instance EC2, le fichier de paire de clés que vous avez créé ou sélectionné lors de la création de l'instance.

`username@instance-id` représente le nom d'utilisateur par défaut pour votre type d'instance et l'ID d'instance, par exemple `ec2-user@i-02573cafcfEXAMPLE` .

> **Tip**
>
> Lorsque vous démarrez une session à l'aide de SSH, vous pouvez copier des fichiers locaux sur l'instance cible en utilisant le format de commande suivant.
>
> ```
> scp -i /path/my-key-pair.pem /path/SampleFile.txt username@instance-id:~
> ```

Pour de plus amples informations sur les autres options possibles avec la commande start-session, veuillez consulter start-session dans le document AWS Systems Manager section of the AWS CLI Command Reference.

Démarrage d'une session (réacheminement de port)

Pour démarrer une session de réacheminement de Session Manager port, la version 2.3.672.0 ou ultérieure de Agent SSM doit être installée sur l'instance gérée.

> **Note**
>
> Avant de démarrer une session, assurez-vous d'avoir effectué les étapes de configuration pour Session Manager. Pour plus d'informations, consultez Configuration d'Session Manager (p. 873).

Pour utiliser l' AWS CLI afin d'exécuter des commandes de session, vous devez installer l' Session Manager plugin sur votre ordinateur local. Pour plus d'informations, consultez la section (Facultatif) Installer le Session Manager plugin pour l'AWS CLI (p. 917).
En fonction de votre système d'exploitation et de votre outil de ligne de commande, le placement des guillemets peut varier et des caractères d'échappement peuvent être requis.

Pour démarrer une session de réacheminement de port, exécutez la commande suivante à partir de l'interface de ligne de commande.

Linux

```
aws ssm start-session \
    --target instance-id \
    --document-name AWS-StartPortForwardingSession \
    --parameters '{"portNumber":["80"], "localPortNumber":["56789"]}'
```

Windows

```
aws ssm start-session ^
    --target instance-id ^
    --document-name AWS-StartPortForwardingSession ^
    --parameters portNumber="3389",localPortNumber="56789"
```

instance-id représente l'ID d'une instance configurée pour être utilisée avec AWS Systems Manager et sa Session Manager capacité, par exemple `i-02573cafcfEXAMPLE` .

portNumber représente le port distant sur l'instance vers laquelle le trafic doit être redirigé, par exemple `3389` . Si ce paramètre n'est pas spécifié, Session Manager présume `80` comme port distant par défaut.

localPortNumber représente le port local sur le client vers lequel le trafic doit être redirigé, par exemple `56789` .

Pour de plus amples informations sur les autres options possibles avec la commande start-session, veuillez consulter start-session dans le document AWS Systems Manager section of the AWS CLI Command Reference.

Pour plus d'informations sur les sessions de réacheminement de port, consultez Réacheminement de port à l'aide d' AWS Systems Manager Session Manager dans le blog AWS News Blog.

Démarrage d'une session (commandes interactives)

Pour démarrer une session de commande interactive, exécutez la commande suivante :

> **Note**
>
> Avant de démarrer une session, assurez-vous d'avoir effectué les étapes de configuration pour Session Manager. Pour plus d'informations, consultez Configuration d'Session Manager (p. 873).
> Pour que vous puissiez utiliser l'AWS CLI pour exécuter des commandes de session, le Session Manager plugin doit également être installé sur votre ordinateur local. Pour plus d'informations, consultez la section (Facultatif) Installer le Session Manager plugin pour l'AWS CLI (p. 917).

```
aws ssm start-session \
    --target instance-id \
    --document-name CustomCommandSessionDocument \
    --parameters '{"logpath":["/var/log/amazon/ssm/amazon-ssm-agent.log"]}'
```

instance-id représente l'ID d'une instance configurée pour être utilisée avec AWS Systems Manager et sa Session Manager capacité, par exemple `i-02573cafcfEXAMPLE` .

Pour de plus amples informations sur les autres options possibles avec la commande start-session, veuillez consulter start-session dans le document AWS Systems Manager section of the AWS CLI Command Reference.

Contenu connexe

Port Forwarding Using AWS Systems Manager Session Manager sur le blog AWS News.

Mettre fin à une session

Vous pouvez utiliser la AWS Systems Manager console ou l' AWS Command Line Interface (AWS CLI) pour mettre fin à une session que vous avez démarrée pour vous connecter à une instance de votre compte. Si aucune activité utilisateur n'a lieu pendant 20 minutes, la session prend fin. Une fois une session terminée, elle ne peut pas être reprise.

Rubriques

- Mise hors service d'une session (console) (p. 928)
- Mise hors service d'une session (AWS CLI) (p. 928)

Mise hors service d'une session (console)

Vous pouvez utiliser la console AWS Systems Manager pour démarrer une session avec une instance de votre compte.

Pour terminer une session (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Session Manager.
3. Pour Sessions, choisissez le bouton d'option situé à gauche de la session que vous souhaitez résilier.
4. Choisissez Terminer.

Mise hors service d'une session (AWS CLI)

Pour mettre fin à une session à l'aide AWS CLI de l' , exécutez la commande suivante.

> **Note**
>
> Pour que vous puissiez utiliser l'AWS CLI pour exécuter des commandes de session, le Session Manager plugin doit également être installé sur votre ordinateur local. Pour de plus amples informations, veuillez consulter (Facultatif) Installer le Session Manager plugin pour l'AWS CLI (p. 917).

```
aws ssm terminate-session \
    --session-id session-id
```

session-id représente l'ID d'une Session Manager session active que vous souhaitez terminer définitivement.

Pour de plus amples informations sur la commande terminate-session, veuillez consulter la page terminate-session dans le AWS Systems Manager section of the AWS CLI Command Reference.

Afficher l'historique de session

Vous pouvez utiliser la AWS Systems Manager console ou l' AWS Command Line Interface (AWS CLI) pour afficher des informations sur les sessions de votre compte. Dans la console, vous pouvez afficher les détails de session suivants :

- L'ID de la session
- Quel utilisateur s'est connecté à une instance au cours de la session
- L'ID de l'instance
- Le début et la fin de la session
- Le statut de la session
- L'emplacement spécifié pour le stockage des journaux de session (si activé)

Avec l'AWS CLI, vous pouvez consulter une liste des sessions dans votre compte, mais pas les informations supplémentaires disponibles dans la console.

Pour en savoir plus sur les détails de l'historique des sessions d'audit et de journalisation, consultez la page Audit et journalisation de l'activité de session (p. 929).

Rubriques

- Afficher l'historique de session (console) (p. 929)
- Affichage de l'historique de session (AWS CLI) (p. 929)

Afficher l'historique de session (console)

Vous pouvez utiliser la console AWS Systems Manager pour afficher des informations sur les sessions de votre compte.

Afficher l'historique de session (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Session Manager.
3. Sélectionnez Start session (Démarrer une session).
4. Sélectionnez l'onglet Session history (Historique de session).

Affichage de l'historique de session (AWS CLI)

Pour afficher une liste des sessions de votre compte à l'aide AWS CLI de l' , exécutez la commande suivante.

```
aws ssm describe-sessions \
    --state History
```

> **Note**
>
> Cette commande renvoie uniquement les résultats pour les connexions à des cibles initiées à l'aide de Session Manager. Il ne répertorie pas les connexions effectuées par d'autres moyens, tels que le protocole RDP (Remote Desktop Protocol) ou le protocole SSH (Secure Shell Protocol).

Pour de plus amples informations sur les autres options possibles avec la commande describe-sessions, veuillez consulter describe-sessions dans le document AWS Systems Manager section of the AWS CLI Command Reference.

Audit et journalisation de l'activité de session

En plus de fournir des informations sur les sessions actuelles et terminées dans la console Systems Manager, Session Manager vous offre plusieurs options d'activité de session de journalisation et d'audit dans votre compte AWS. Cela vous permet d'effectuer les tâches suivantes :

- Créer et stocker des journaux de session à des fins d'archivage.
- Générer un rapport affichant les détails de chaque connexion à vos instances via Session Manager au cours des 30 derniers jours.
- Générer des notifications d'activité de session sur votre compte AWS, telles que des notifications Amazon Simple Notification Service (Amazon SNS).
- Lancer automatiquement une autre action sur une ressource AWS à la suite de l'activité de session, par exemple l'exécution d'une fonction AWS Lambda, le démarrage d'un pipeline AWS CodePipeline ou l'exécution d'un document AWS Systems Manager Fonctionnalité Exécuter la commande.

> **Important**
>
> Prenez note des exigences et limitations suivantes pour Session Manager :
>
> - Si vous utilisez Windows Server 2012 ou version antérieure, les données contenues dans vos journaux peuvent ne pas être formatées de manière optimale. Nous vous recommandons d'utiliser Windows Server 2012 R2 et versions ultérieures pour optimiser vos formats de journaux.
> - Si vous utilisez des instances Linux, assurez-vous que screen L'utilitaire est installé. Si ce n'est pas le cas, il se peut que vos données de journaux soient tronquées. Sur Amazon Linux , et Amazon Linux 2 , leUbuntu Server screen L'utilitaire est installé par défaut. Pour installer screen exécutez manuellement , en fonction de votre version de Linux, ou `sudo yum install screen` .`sudo apt-get install screen`
> - La journalisation et l'audit ne sont pas disponibles pour les sessions Session Manager qui se connectent via le transfert de port ou SSH. Cela est dû au fait que SSH crypte toutes les données de session et que Session Manager sert uniquement de tunnel pour les connexions SSH.

Pour plus d'informations sur les autorisations requises pour utiliser Amazon S3 ou Amazon CloudWatch Logs pour l'enregistrement des données de session, consultez Création d'un profil d'instance avec les autorisations pour Session Manager, Amazon S3 et CloudWatch Logs (p. 881).

Pour de plus amples informations sur les options de journalisation et d'audit pour Session Manager, veuillez consulter les rubriques suivantes.

Rubriques

- Auditer l'activité de session avec AWS CloudTrail (p. 930)
- Journalisation des données de session avec Amazon S3 (console) (p. 931)
- Journalisation des données de session avec Amazon CloudWatch Logs (console) (p. 932)
- Surveillance de l'activité de la session avec Amazon EventBridge (console) (p. 933)

Auditer l'activité de session avec AWS CloudTrail

AWS CloudTrailcapture les appels d'API de session via la Systems Manager console , l' AWS Command Line Interface (AWS CLI) et le Systems Manager kit SDK . Vous pouvez afficher les informations sur la CloudTrail console ou les stocker dans un compartiment Amazon Simple Storage Service (Amazon S3) spécifié. Un compartiment S3 est utilisé pour tous les CloudTrail journaux de votre compte.

Pour plus d'informations, consultez la section Journalisation des appels d'API AWS Systems Manager avec AWS CloudTrail (p. 1336).

Journalisation des données de session avec Amazon S3 (console)

Vous pouvez choisir de stocker les données des journaux de session dans un compartiment S3 de votre choix à des fins d'audit. L'option par défaut est pour les journaux à envoyer à un compartiment S3 chiffré. Le chiffrement est effectué à l'aide de la clé spécifiée pour le compartiment, soit une clé AWS Key Management Service (AWS KMS) ou une clé de chiffrement côté serveur Amazon S3 (SSE) (AES-256).

> **Important**
>
> Lorsque vous utilisez des compartiments d'hébergement virtuel avec SSL (Secure Sockets Layer), le certificat générique SSL correspond uniquement aux compartiments qui ne contiennent pas de points. Pour contourner ce problème, utilisez HTTP ou écrivez votre propre logique de vérification de certificat. Il est recommandé de ne pas utiliser de point (".") dans les noms de compartiment lors de l'utilisation de compartiments d'hébergement virtuel.

Chiffrement de compartiment S3

Pour envoyer les journaux vers votre compartiment S3 via le chiffrement, le chiffrement doit être activé sur le compartiment. Pour de plus amples informations sur le chiffrement de compartiments S3, veuillez consulter la page Chiffrement Amazon S3 par défaut pour les compartiments S3.

Clé CMK gérée par le client

Si vous utilisez une clé principale AWS KMS client (CMK) que vous gérez vous-même (une clé CMK gérée par le client) pour chiffrer votre compartiment, le profil IAM d'instance attaché à vos instances doit disposer d'autorisations explicites pour lire la clé CMK. Si vous utilisez une clé CMK gérée par AWS, l'instance ne nécessite pas d'autorisation explicite. Pour de plus amples informations sur l'octroi d'une autorisation d'utilisation d'une clé CMK à un profil d'instance, veuillez consulter Autorise les utilisateurs de clé à utiliser la clé CMK dans le AWS Key Management Service Developer Guide.

Suivez ces étapes pour configurer Session Manager afin de stocker les journaux de session dans un compartiment Amazon S3.

> **Note**
>
> Vous pouvez également utiliser l'AWS CLI pour spécifier ou modifier le compartiment S3 vers lequel les données de session sont envoyées. Pour plus d'informations, consultez la section Mettre à jour les préférences Session Manager (ligne de commande) (p. 906).

Pour consigner les données de session en utilisant Amazon S3 (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Session Manager.
3. Sélectionnez l'onglet Préférences, puis Modifier.
4. Cochez la case en regard de S3 bucket (Compartiment S3).
5. (Facultatif) Si vous ne voulez pas chiffrer les données de journaux qui sont envoyées vers le compartiment S3, cochez la case en regard de Encrypt log data (Chiffrer les données de journaux). Sinon, les données de journaux sont chiffrées à l'aide de la clé de chiffrement côté serveur spécifiée pour le compartiment. Vous devez également désactiver la case à cocher si le chiffrement n'est pas activé sur le compartiment S3.
6. Pour S3 bucket name (Nom du compartiment S3), sélectionnez l'une des opérations suivantes :

   > **Note**
   >
   > Il est recommandé de ne pas utiliser de point (".") dans les noms de compartiment lors de l'utilisation de compartiments d'hébergement virtuel. Pour plus d'informations sur

les conventions de dénomination de compartiment S3, consultez Limites et restrictions applicables aux compartiments dans le Amazon Simple Storage Service Manuel du développeur.

- Choisissez un nom de compartiment dans la liste : Sélectionnez un compartiment S3 qui a déjà été créé dans votre compte pour stocker les données des journaux de session.
- Saisissez un nom de compartiment dans la zone de texte : Saisissez le nom d'un compartiment S3 qui a déjà été créé dans votre compte pour stocker les données de journaux de session.

7. (Facultatif) Pour S3 key prefix (Préfixe de clé S3), saisissez le nom d'un dossier existant ou d'un nouveau dossier pour stocker les journaux dans le compartiment sélectionné.
8. Choisissez Save.

Pour de plus amples informations sur l'utilisation d'Amazon S3 et des compartiments S3, veuillez consulter les guides Amazon Simple Storage Service Guide de mise en route et Amazon Simple Storage Service Guide de l'utilisateur de la console.

Journalisation des données de session avec Amazon CloudWatch Logs (console)

Avec Amazon CloudWatch Logs , vous pouvez surveiller, stocker et consulter les fichiers journaux de différents AWS services . Vous pouvez envoyer des données de journaux de session à un groupe de journaux CloudWatch Logs à des fins d'audit. L'option par défaut est que les données de journaux soient envoyées avec chiffrement à l'aide de votre clé AWS KMS, mais vous pouvez envoyer les données à votre groupe de journaux avec ou sans chiffrement.

Suivez ces étapes pour configurer Session Manager afin d'envoyer des données de journaux de session à un groupe de journaux CloudWatch Logs Logs.

> **Note**
>
> Vous pouvez également utiliser l'AWS CLI pour spécifier ou modifier le groupe de journaux CloudWatch Logs vers lequel les données de session sont envoyées. Pour plus d'informations, consultez la section Mettre à jour les préférences Session Manager (ligne de commande) (p. 906).

Pour consigner les données de session en utilisant Amazon CloudWatch Logs (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Session Manager.
3. Sélectionnez l'onglet Préférences, puis Modifier.
4. Cochez la case en regard de CloudWatch logs (Journaux CloudWatch).
5. (Facultatif) Si vous ne souhaitez pas chiffrer les données de journaux qui sont envoyées vers CloudWatch Logs Logs, cochez la case en regard de Encrypt log data (Chiffrer les données de journaux). Sinon, les données de journaux sont chiffrées à l'aide du chiffrement côté serveur des clés spécifiées pour le groupe de journaux. Vous devez également désactiver la case à cocher si le chiffrement n'est pas activé sur le groupe de journaux.
6. Pour CloudWatch logs (Journaux CloudWatch), sélectionnez l'une des actions suivantes pour spécifier le groupe de journaux CloudWatch Logs dans votre compte AWS où charger les journaux de session :

   - Choisissez un groupe de journaux dans la liste : Sélectionnez un groupe de journaux qui a déjà été créé dans votre compte pour stocker les données de journaux de session.
   - Saisissez un nom de groupe de journaux dans la zone de texte : Saisissez le nom d'un groupe de journaux qui a déjà été créé dans votre compte pour stocker les données de journaux de session.

7. Choisissez Save.

Pour de plus amples informations sur l'utilisation de CloudWatch Logs, consultez Amazon CloudWatch Logs User Guide.

Surveillance de l'activité de la session avec Amazon EventBridge (console)

Avec EventBridge , vous pouvez configurer des règles pour détecter d'éventuelles modifications des AWS ressources . Vous pouvez créer une règle pour détecter lorsqu'un utilisateur de votre organisation démarre ou met fin à une session puis, par exemple, recevoir une notification via Amazon SNS vous informant de l'événement.

La prise en charge de EventBridge pour Session Manager repose sur les enregistrements des actions d'API qui ont été enregistrées par CloudTrail. (Vous pouvez utiliser l'intégration CloudTrail avec EventBridge pour répondre à la plupart des événements AWS Systems Manager.)

Les étapes suivantes décrivent comment lancer des notifications via Amazon Simple Notification Service (Amazon SNS) lorsquSession Manager'un événement d'API se produit, par exemple StartSession .

Pour surveiller l'activité de session à l'aide de Amazon EventBridge (console)

1. Créez une rubrique Amazon SNS à utiliser pour envoyer des notifications lorsque l'événement Session Manager que vous souhaitez suivre se produit.

   Pour plus d'informations, consultez Création d'une rubrique dans le Amazon Simple Notification Service Manuel du développeur.
2. Créez une EventBridge règle pour appeler la Amazon SNS cible pour le type d'Session Managerévénement que vous souhaitez suivre.

   Pour plus d'informations sur la façon de créer la règle, consultez Création d'une EventBridge règle qui se déclenche sur un événement à partir d'une ressource AWS dans le Guide de l'utilisateur Amazon EventBridge .

   Tout au long de la création de votre règle, sélectionnez les éléments suivants :

   - Pour Service Name (Nom du service), sélectionnez EC2 Simple Systems Manager (SSM) (SSM EC2).
   - Pour Event Type (Type d'événement), sélectionnez AWS API Call via CloudTrail (Appel d'API AWS via CloudTrail).
   - Sélectionnez Specific operation(s) (Opérations spécifiques), puis saisissez la ou les commandes Session Manager (l'une après l'autre) pour lesquelles vous souhaitez recevoir des notifications. Vous pouvez choisir StartSession , et ResumeSession (TerminateSession ne prend pas en charge EventBridge les `Get*` commandes , et .) `List*Describe*`
   - Pour Targets (Cibles), choisissez la rubrique SNS. Pour Rubrique, choisissez le nom de la rubrique Amazon SNS que vous avez créée à l'étape 1.

Pour plus d'informations, consultez le Guide de l'utilisateur Amazon EventBridge et le Amazon Simple Notification Service Guide de mise en route.

Schéma du document de session

Les informations suivantes décrivent les éléments de schéma d'un document de session. Session Manager utilise des documents de session pour déterminer le type de session à démarrer, par exemple une

session standard, une session de réacheminement de port ou une session pour exécuter une commande interactive.

schemaVersion (p. 936)

Version de schéma du document de session. Actuellement, les documents de session prennent uniquement en charge la version 1.0.

Type: String (chaîne)

Obligatoire : Oui

description (p. 936)

Description que vous spécifiez pour le document Session. Par exemple, « Document to start port forwarding session with Session Manager ".

Type: String (chaîne)

Obligatoire : Non

sessionType (p. 936)

Le type de session que le document Session est utilisé pour établir.

Type: String (chaîne)

Obligatoire : Oui

Valeurs valides : `InteractiveCommands`| | |`PortStandard_Stream`

inputs (p. 936)

Préférences de session à utiliser pour les sessions établies à l'aide de ce document de session. Cet élément est obligatoire pour les documents de session qui sont utilisés pour créer `Standard_Stream` des sessions.

Type: StringMap

Obligatoire : Non

s3BucketName (p. 936)

Le compartiment Amazon Simple Storage Service (Amazon S3) auquel vous souhaitez envoyer des journaux de session à la fin de vos sessions.

Type: String (chaîne)

Obligatoire : Non

s3KeyPrefix (p. 936)

Préfixe à utiliser lors de l'envoi des journaux au compartiment S3 que vous avez spécifié dans l'`s3BucketName`entrée. Pour plus d'informations sur l'utilisation d'un préfixe partagé avec des objets stockés dans Amazon S3 , consultez Comment utiliser des dossiers dans un compartiment S3 ? dans le Amazon Simple Storage Service Guide de l'utilisateur de la console .

Type: String (chaîne)

Obligatoire : Non

s3EncryptionEnabled (p. 936)

Si la valeur est définie, le compartiment S3 que vous avez spécifié dans l`true`'entrée doit être chiffré.`s3BucketName`

Type: Booléen

Obligatoire : Oui

cloudWatchLogGroupName (p. 937)

Le nom du groupe Amazon CloudWatch Logs (CloudWatch Logs) auquel vous souhaitez envoyer des journaux de session à la fin de vos sessions.

Type: String (chaîne)

Obligatoire : Non

cloudWatchEncryptionEnabled (p. 937)

Si ce paramètre est défini sur `true` , le groupe de journaux que vous avez spécifié dans l`cloudWatchLogGroupName`'entrée doit être chiffré.

Type: Booléen

Obligatoire : Oui

kmsKeyId (p. 937)

L'ID de la clé AWS Key Management Service (AWS KMS) que vous souhaitez utiliser pour chiffrer davantage les données entre vos ordinateurs clients locaux et les instances Amazon Elastic Compute Cloud (Amazon EC2) auxquelles vous vous connectez.

Type: String (chaîne)

Obligatoire : Non

runAsEnabled (p. 937)

Si cette option est définie sur `true` , vous devez spécifier un compte utilisateur qui existe sur les instances auxquelles vous vous connecterez dans l'`runAsDefaultUser`entrée. Dans le cas contraire, le démarrage des sessions échouera. Par défaut, les sessions sont démarrées à l'aide du `ssm-user` compte créé par l' Agent SSM . La fonction Exécuter en tant que est uniquement prise en charge pour la connexion aux instances Linux.

Type: Booléen

Obligatoire : Oui

runAsDefaultUser (p. 937)

Nom du compte utilisateur avec lequel démarrer les sessions sur les instances Linux lorsque l'`runAsEnabled`entrée est définie sur `true` . Le compte utilisateur que vous spécifiez pour cette entrée doit exister sur les instances auxquelles vous allez vous connecter ; sinon, les sessions ne parviendront pas à démarrer.

Type: String (chaîne)

Obligatoire : Non

shellProfile (p. 937)

Les préférences que vous spécifiez par système d'exploitation à appliquer dans les sessions, telles que les préférences shell, les variables d'environnement, les répertoires de travail et l'exécution de plusieurs commandes lorsqu'une session est démarrée.

Type: StringMap

Obligatoire : Non

windows (p. 937)

Les préférences shell, les variables d'environnement, les répertoires de travail et les commandes que vous spécifiez pour les sessions sur des instances Windows.

Type: String (chaîne)

Obligatoire : Non

linux (p. 937)

Les préférences shell, les variables d'environnement, les répertoires de travail et les commandes que vous spécifiez pour les sessions sur les instances Linux.

Type: String (chaîne)

Obligatoire : Non

parameters (p. 937)

Objet qui définit les paramètres acceptés par le document. Pour plus d'informations sur la définition des paramètres de document, consultez les paramètres dans le Éléments de niveau supérieur (p. 1227) . Pour les paramètres que vous référencez souvent, nous vous recommandons de les stocker dans Systems Manager Parameter Store, puis de les référencer. Vous pouvez référencer les paramètres Parameter Store `String` et `StringList` dans cette section d'un document. Vous pouvez référencer les paramètres Parameter Store `SecureString` dans cette section d'un document. Vous pouvez référencer un Parameter Store paramètre à l'aide du format suivant.

```
{{ssm:parameter-name}}
```

Pour plus d'informations sur Parameter Store, consultez AWS Systems Manager Parameter Store (p. 218).

Type: StringMap

Obligatoire : Non

properties (p. 937)

Objet dont les valeurs que vous spécifiez sont utilisées dans l'appel `StartSession` d'API.

Pour les documents de session utilisés pour `InteractiveCommands` les sessions, l'objet des propriétés inclut les commandes à exécuter sur les systèmes d'exploitation que vous spécifiez. Pour plus d'informations, consultez la section (Facultatif) Restriction de l'accès aux commandes dans une session (p. 908).

Pour les documents de session utilisés pour `Port` les sessions, l'objet des propriétés contient le numéro de port vers lequel le trafic doit être redirigé. Pour obtenir un exemple, consultez l'exemple de document de `Port` type Session ci-dessous.

Type: StringMap

Obligatoire : Non

`Standard_Stream`Exemple de document de type Session

YAML

```
---
schemaVersion: '1.0'
description: Document to hold regional settings for Session Manager
sessionType: Standard_Stream
inputs:
  s3BucketName: ''
  s3KeyPrefix: ''
  s3EncryptionEnabled: true
```

```
  cloudWatchLogGroupName: ''
  cloudWatchEncryptionEnabled: true
  kmsKeyId: ''
  runAsEnabled: true
  runAsDefaultUser: ''
  shellProfile:
    windows: ''
    linux: ''
```

JSON

```
{
    "schemaVersion": "1.0",
    "description": "Document to hold regional settings for Session Manager",
    "sessionType": "Standard_Stream",
    "inputs": {
        "s3BucketName": "",
        "s3KeyPrefix": "",
        "s3EncryptionEnabled": true,
        "cloudWatchLogGroupName": "",
        "cloudWatchEncryptionEnabled": true,
        "kmsKeyId": "",
        "runAsEnabled": true,
        "runAsDefaultUser": "",
        "shellProfile": {
            "windows": "date",
            "linux": "pwd;ls"
        }
    }
}
```

`InteractiveCommands`Exemple de document de type Session

YAML

```
---
schemaVersion: '1.0'
description: Document to view a log file on a Linux instance
sessionType: InteractiveCommands
parameters:
  logpath:
    type: String
    description: The log file path to read.
    default: "/var/log/amazon/ssm/amazon-ssm-agent.log"
    allowedPattern: "^[a-zA-Z0-9-_/]+(.log)$"
properties:
  linux:
    commands: "tail -f {{ logpath }}"
    runAsElevated: true
```

JSON

```
{
    "schemaVersion": "1.0",
    "description": "Document to view a log file on a Linux instance",
    "sessionType": "InteractiveCommands",
    "parameters": {
        "logpath": {
            "type": "String",
            "description": "The log file path to read.",
            "default": "/var/log/amazon/ssm/amazon-ssm-agent.log",
```

```
            "allowedPattern": "^[a-zA-Z0-9-_/]+(.log)$"
        }
    },
    "properties": {
        "linux": {
            "commands": "tail -f {{ logpath }}",
            "runAsElevated": true
        }
    }
}
```

`Port`Exemple de document de type Session

YAML

```
---
schemaVersion: '1.0'
description: Document to open given port connection over Session Manager
sessionType: Port
parameters:
  paramExample:
    type: string
    description: document parameter
properties:
  portNumber: anyPortNumber
```

JSON

```
{
    "schemaVersion": "1.0",
    "description": "Document to open given port connection over Session Manager",
    "sessionType": "Port",
    "parameters": {
        "paramExample": {
            "type": "string",
            "description": "document parameter"
        }
    },
    "properties": {
        "portNumber": "anyPortNumber"
    }
}
```

Dépannage de Session Manager

Consultez les informations suivantes pour tenter de résoudre les problèmes liés à Session Manager.

Rubriques

- Je n'ai pas le droit de démarrer une session (p. 939)
- Je n'ai pas le droit de modifier les préférences de session (p. 939)
- L'instance n'est pas disponible ou n'est pas configurée pour Session Manager (p. 939)
- Session Manager plugin introuvable (p. 940)
- Session Manager plugin n'est pas ajouté automatiquement au chemin de la ligne de commande (Windows) (p. 940)
- TargetNotConnected (p. 941)
- Affichage d'un écran vide après le démarrage d'une session (p. 941)

Je n'ai pas le droit de démarrer une session

Problème : Vous essayez de démarrer une session, mais le système vous indique que vous ne disposez pas des autorisations nécessaires.

- Solution: Un administrateur système ne vous a pas accordé les autorisations AWS Identity and Access Management de stratégie (IAM) pour démarrer Session Manager des sessions. Pour plus d'informations, consultez la section Contrôler les accès de session utilisateur aux instances (p. 882).

Je n'ai pas le droit de modifier les préférences de session

Problème : Vous essayez de mettre à jour les préférences de session globales pour votre organisation, mais le système vous indique que vous ne disposez pas des autorisations nécessaires.

- Solution: Un administrateur système ne vous a pas accordé les autorisations de IAM stratégie pour définir Session Manager des préférences. Pour plus d'informations, consultez la section Accorder ou révoquer des autorisations utilisateur pour mettre à jour des préférences Session Manager (p. 898).

L'instance n'est pas disponible ou n'est pas configurée pour Session Manager

Problème 1 : Vous souhaitez démarrer une session sur la page Démarrer une console de session, mais une instance ne figure pas dans la liste.

- Solution: L'instance à laquelle vous souhaitez vous connecter n'a peut-être pas été configurée pour être utilisée avec le AWS Systems Manager service. Pour utiliser une instance avec Systems Manager, un profil d'instance IAM accordant l'autorisation Systems Manager d'effectuer des actions sur vos instances doit être attaché à l'instance. Pour plus d'informations, consultez la section Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34).

  Note

  Si Agent SSM est déjà en cours d'exécution sur une instance lorsque vous attachez le profil IAM d'instance, vous devrez peut-être redémarrer l'agent avant que l'instance ne soit répertoriée sur la page Démarrer une console de session.

Problème 2 : Une instance que vous souhaitez connecter figure dans la liste sur la page Start a session console (Démarrer une console de session), mais la page indique que « L'instance que vous avez sélectionnée n'est pas configurée pour utiliser Session Manager . »

- Solution A : L'instance a été configurée pour être utilisée avec le AWS Systems Manager service , mais le profil IAM d'instance attaché à l'instance peut ne pas inclure les autorisations pour la Session Manager capacité. Pour en savoir plus, consultez la page Vérifier ou créer un profil d'instance IAM avec des autorisations Session Manager (p. 876).
- Solution B : L'instance n'exécute pas une version d' Agent SSM qui prend en charge Session Manager . Mettez à jour l'Agent SSM de l'instance vers la version 2.3.68.0 ou ultérieure.

  Mettez à jour l'Agent SSM manuellement sur une instance en suivant la procédure décrite sur les pages Installation manuelle Agent SSM sur les instances EC2 pourWindows Server (p. 100) ou Installer manuellement l'Agent SSM sur les instances EC2 pour Linux (p. 76), selon votre système d'exploitation.

  Vous pouvez également utiliser le document Fonctionnalité Exécuter la commande `AWS-UpdateSSMAgent` pour mettre à jour la version de l'agent sur une ou plusieurs instances à la fois. Pour plus d'informations, consultez la section Mettre à jour l'Agent SSM à l'aide de Fonctionnalité Exécuter la commande (p. 948).

Tip

Pour garder constamment votre agent à jour, nous vous recommandons de configurer la mise à jour automatique de l'Agent SSM vers la dernière version, tel qu'expliqué ci-dessous :

- Exécutez `AWS-UpdateSSMAgent` dans le cadre d'une association State Manager. Pour plus d'informations, consultez la section Mise à jour automatique d'Agent SSM (interface de ligne de commande) (p. 1043).
- Exécutez `AWS-UpdateSSMAgent` dans le cadre d'une fenêtre de maintenance. Pour de plus amples informations sur l'utilisation des fenêtres de maintenance, veuillez consulter Utilisation des fenêtres de maintenance (console) (p. 707) et Didacticiel : Créer et configurer une fenêtre de maintenance (AWS CLI) (p. 717).

- Solution C : L'instance se trouve dans un Virtual Private Cloud (VPC), mais aucun `ssmmessages` point de terminaison n'a été créé dans le VPC. Un `ssmmessages` point de terminaison au format com.amazonaws.***region***.ssmmessages est requis si vous vous connectez à vos instances via un canal de données sécurisé à l'aide d' Session Manager . Pour plus d'informations, consultez la section Création des points de terminaison d'un VPC pour Systems Manager (p. 44).
- Solution D : L'instance a des ressources d'UC ou de mémoire disponibles limitées. Même si votre instance peut autrement être fonctionnelle, si elle ne dispose pas de ressources suffisantes, vous ne pouvez pas établir de session. Pour de plus amples informations, veuillez consulter Résolution d'un problème d'instance inaccessible.

Session Manager plugin introuvable

Pour que vous puissiez utiliser l'AWS CLI pour exécuter des commandes de session, le Session Manager plugin doit également être installé sur votre ordinateur local. Pour de plus amples informations, veuillez consulter (Facultatif) Installer le Session Manager plugin pour l'AWS CLI (p. 917).

Session Manager plugin n'est pas ajouté automatiquement au chemin de la ligne de commande (Windows)

Lorsque vous installez le Session Manager plugin sur Windows, le fichier exécutable `session-manager-plugin` doit être ajouté automatiquement à la variable d'environnement `PATH` de votre système d'exploitation. Si la commande échoue après l'avoir exécutée pour vérifier si le Session Manager plugin était correctement installé (`aws ssm start-session --target` *`instance-id`*), vous devrez peut-être effectuer cet ajout manuellement à l'aide de la procédure suivante.

Pour modifier votre variable PATH (Windows)

1. Appuyez sur la touche Windows et entrez **`environment variables`**.
2. Choisissez Modifier les variables d'environnement pour votre compte.
3. Choisissez PATH, puis Modifier.
4. Ajoutez des chemins au champ de valeur de variable, séparés par des points-virgules, comme illustré dans cet exemple : *`chemin\\C:\\existant`*;***`C:\\nouveau\\chemin`***

   `chemin\\C:\\existant` représente la valeur déjà dans le champ. ***`C:\\nouveau\\chemin`*** représente le chemin que vous souhaitez ajouter, comme illustré dans ces exemples.

   - machines 64 bits : `C:\Program Files\Amazon\SessionManagerPlugin\bin\`
   - machines 32 bits : `C:\Program Files (x86)\Amazon\SessionManagerPlugin\bin\`
5. Choisissez OK deux fois pour appliquer les nouveaux paramètres.
6. Fermez toute invite de commande en cours d'exécution et rouvrez.

TargetNotConnected

Problème : Vous essayez de démarrer une session, mais le système renvoie le message d'erreur, « Une erreur s'est produite (TargetNotConnected) lors de l'appel de l'StartSessionopération : *`InstanceID`* n'est pas connecté. »

- Solution: Cette erreur est renvoyée lorsque l'instance cible spécifiée pour la session n'est pas entièrement configurée pour une utilisation avec le Gestionnaire de session. Pour plus d'informations, consultez la section Configuration d'Session Manager (p. 873).

Affichage d'un écran vide après le démarrage d'une session

Problème : Vous démarrez une session et Session Manager affichez un écran vide.

- Solution A : Ce problème peut se produire lorsque le volume racine de l'instance est plein. En raison du manque d'espace disque, l'Agent SSM sur l'instance cesse de fonctionner. Pour résoudre ce problème, utilisez Amazon CloudWatch pour collecter des métriques et des journaux à partir des systèmes d'exploitation. Pour plus d'informations, consultez Surveillance des métriques de mémoire et de disque pour les instances Amazon EC2 Linux ou Surveillance des métriques de mémoire et de disque pour les instances Amazon EC2 Windows.
- Solution B : Un écran vide peut s'afficher si vous avez accédé à la console à l'aide d'un lien qui inclut un point de terminaison et une paire de régions non appariées. Par exemple, dans l'URL de la console suivante, `us-west-2` est le point de terminaison spécifié, mais `us-west-1` est la AWS région spécifiée.

```
https://us-west-2.console.aws.amazon.com/systems-manager/session-manager/sessions?
region=us-west-1
```

- Solution C : L'instance se connecte à à Systems Manager l'aide de points de terminaison de VPC et vos Session Manager préférences écrivent la sortie de session dans un Amazon S3 compartiment , mais aucun point de terminaison de `s3` passerelle n'existe dans le VPC. Un `s3` point de terminaison au format com.amazonaws.*`region`*.s3 est requis si vos instances se connectent à à Systems Manager l'aide de points de terminaison de VPC et que vos Session Manager préférences écrivent la sortie de session dans un Amazon S3 compartiment . Pour plus d'informations, consultez la section Création des points de terminaison d'un VPC pour Systems Manager (p. 44).
- Solution D : Le groupe de journaux ou le compartiment S3 que vous avez spécifié dans vos préférences de session a été supprimé. Pour résoudre ce problème, mettez à jour vos préférences de session avec un groupe de journaux ou un compartiment S3 valide.
- Solution E : Le groupe de journaux ou le compartiment S3 que vous avez spécifié dans vos préférences de session n'est pas chiffré, mais vous avez défini le `cloudWatchEncryptionEnabled` ou l`s3EncryptionEnabled`'entrée sur `true` . Pour résoudre ce problème, mettez à jour vos préférences de session avec un groupe de journaux ou un compartiment S3 chiffré, ou définissez l'`cloudWatchEncryptionEnabled`entrée `s3EncryptionEnabled` ou sur `false` . Ce scénario s'applique uniquement aux clients qui créent des préférences de session à l'aide d'outils de ligne de commande.

AWS Systems Manager Fonctionnalité Exécuter la commande

AWS Systems Manager Fonctionnalité Exécuter la commande permet de gérer, à distance et en toute sécurité, la configuration de vos instances gérées. Une instance gérée est toute instance EC2 ou machine sur site de votre environnement hybride ayant été configurée pour Systems Manager. Avec la fonctionnalité Exécuter la commande, vous pouvez automatiser les tâches administratives courantes et effectuer les changements de configuration opportuns à grande échelle. Vous pouvez utiliser Fonctionnalité Exécuter

la commande à partir de la console AWS, l'AWS Command Line Interface, les Outils AWS pour Windows PowerShell ou les kits de développement logiciel (SDK) AWS. Fonctionnalité Exécuter la commande est proposé sans frais supplémentaires.

Les administrateurs ont recours à la fonctionnalité Fonctionnalité Exécuter la commande pour effectuer les types de tâches suivants sur leurs instances gérées : installer ou amorcer des applications, créer un pipeline de déploiement, capturer des fichiers journaux lorsqu'une instance est mise hors service à partir d'un groupe Auto Scaling et joindre des instances à un domaine Windows, pour n'en citer que quelques-uns.

Démarrer

Le tableau suivant comporte des informations pour vous aider à vous familiariser avec la fonctionnalité Exécuter la commande.

Rubrique	Details (Détails)
Prérequis d'Systems Manager (p. 15)	(Obligatoire) Vérifiez que vos instances respectent la configuration minimale requise pour la fonctionnalité Fonctionnalité Exécuter la commande, configurez les rôles nécessaires et installez l'Agent SSM.
Configuration d'AWS Systems Manager pour les environnements hybrides (p. 49)	(Facultatif) Enregistrez les serveurs sur site et VMs auprès d'AWS afin de pouvoir les gérer à l'aide d' Fonctionnalité Exécuter la commande .
Exécuter des commandes avec Systems Manager Fonctionnalité Exécuter la commande (p. 946)	Découvrez comment exécuter une commande à partir de la console EC2 et exécuter des commandes sur un parc d'instances gérées.
Procédures Fonctionnalité Exécuter la commande (p. 970)	Découvrez comment exécuter des commandes avec les Outils AWS pour Windows PowerShell ou l'AWS CLI.

Prise en charge de Amazon EventBridge

Cette fonction Systems Manager est prise en charge en tant que type d'événement et type de cible dans les règles EventBridge. Pour plus d'informations, consultez Surveillance des événements Systems Manager avec Amazon EventBridge (p. 1340) et Référence : modèles et types Amazon EventBridge d'événements pourSystems Manager (p. 1387).

Contenu connexe

- Exécution des commandes à distance sur une instance EC2 (Didacticiel de 10 minutes)
- Quotas de service Systems Manager dans le Référence générale d'Amazon Web Services.
- Référence d'API AWS Systems Manager

Table des matières

- Configuration d'Fonctionnalité Exécuter la commande (p. 943)
- Exécuter des commandes avec Systems Manager Fonctionnalité Exécuter la commande (p. 946)
- Gestion des codes de sortie avec des scripts (p. 963)
- Comprendre les états des commandes (p. 965)
- Procédures Fonctionnalité Exécuter la commande (p. 970)
- Dépannage de la fonctionnalité Exécuter la commande de Systems Manager (p. 986)

Configuration d'Fonctionnalité Exécuter la commande

Avant de pouvoir gérer des instances à l'aide de Fonctionnalité Exécuter la commande, vous devez configurer une stratégie utilisateur AWS Identity and Access Management (IAM) pour tout utilisateur qui exécutera des commandes. Pour plus d'informations, consultez la section Créer des utilisateurs et des groupes IAM non administrateurs pour Systems Manager (p. 28).

Vous devez également créer un rôle de profil d'instance IAM pour toutes les instances qui traiteront des commandes et l'attacher à ces instances. Pour de plus amples informations, veuillez consulter Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34) et Attachement d'un profil d'instance IAM à une instance EC2 (p. 40).

Nous vous recommandons également d'effectuer les tâches de configuration facultatives suivantes afin de minimiser le dispositif de sécurité et la gestion quotidienne de vos instances.

Surveiller les exécutions de commandes à l'aide d'Amazon EventBridge

Vous pouvez utiliser Amazon EventBridge pour consigner les changements de statut d'exécution des commandes. Vous pouvez créer une règle qui s'exécute à chaque changement de statut ou lorsqu'un ou plusieurs statuts spécifiques sont activés. Vous pouvez également spécifier Fonctionnalité Exécuter la commande en tant qu'action cible lorsquEventBridge'un événement se produit. Pour plus d'informations, consultez la section Configuration de EventBridge pour Systems Manager événements (p. 1341).

Surveiller les exécutions de commandes à l'aide d'Amazon CloudWatch Logs

Vous pouvez configurer Fonctionnalité Exécuter la commande pour envoyer régulièrement tous les journaux de sortie de commande et d'erreurs à un groupe de journaux CloudWatch Logs. Vous pouvez surveiller ces journaux de sortie quasiment en temps réel, rechercher des phrases, valeurs ou modèles spécifiques, et créer des alarmes en fonction de la recherche. Pour plus d'informations, consultez la section Configuration d'Amazon CloudWatch Logs pour Fonctionnalité Exécuter la commande (p. 1338).

Restreindre l'accès de la fonctionnalité Fonctionnalité Exécuter la commande à des instances spécifiques

Vous pouvez restreindre les instances gérées sur lesquelles des commandes peuvent être exécutées en créant une stratégie d'utilisateur IAM qui comporte une condition selon laquelle l'utilisateur ne peut exécuter de commandes que sur les instances comportant des balises Amazon EC2 spécifiques. Pour de plus amples informations, veuillez consulter la rubrique Restriction de l'accès de la fonctionnalité Fonctionnalité Exécuter la commande en fonction des balises d'instance (p. 943).

Restriction de l'accès de la fonctionnalité Fonctionnalité Exécuter la commande en fonction des balises d'instance

Vous pouvez restreindre l'exécution de commandes à des instances spécifiques en créant une stratégie d'utilisateur IAM qui comporte une condition selon laquelle l'utilisateur ne peut exécuter de commandes que sur les instances comportant des balises Amazon EC2 spécifiques. Dans l'exemple suivant, l'utilisateur est autorisé à utiliser Fonctionnalité Exécuter la commande (`Effect: Allow, Action: ssm:SendCommand`) à l'aide de n'importe quel SSM document (`Resource: arn:aws:ssm:*:*:document/*`) sur n'importe quelle instance (`Resource: arn:aws:ec2:*:*:instance/*`) à condition que l'instance soit une Finance WebServer (`ssm:resourceTag/Finance: WebServer`). Si l'utilisateur envoie une commande à une instance non balisée ou qui possède une balise autre que `Finance: WebServer`, les résultats d'exécution indiquent `AccessDenied`.

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

```
      {
         "Effect":"Allow",
         "Action":[
            "ssm:SendCommand"
         ],
         "Resource":[
            "arn:aws:ssm:*:*:document/*"
         ]
      },
      {
         "Effect":"Allow",
         "Action":[
            "ssm:SendCommand"
         ],
         "Resource":[
            "arn:aws:ec2:*:*:instance/*"
         ],
         "Condition":{
            "StringLike":{
               "ssm:resourceTag/Finance":[
                  "WebServers"
               ]
            }
         }
      }
   ]
}
```

Vous pouvez créer des stratégies IAM qui permettent à un utilisateur d'exécuter des commandes sur des instances balisées à l'aide de plusieurs balises. La stratégie suivante permet à l'utilisateur d'exécuter des commandes sur des instances dotées de deux balises. Si un utilisateur envoie une commande à une instance non balisée à l'aide de ces deux balises, les résultats d'exécution indiquent `AccessDenied`.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "ssm:SendCommand"
         ],
         "Resource":"*",
         "Condition":{
            "StringLike":{
               "ssm:resourceTag/tag_key1":[
                  "tag_value1"
               ],
               "ssm:resourceTag/tag_key2":[
                  "tag_value2"
               ]
            }
         }
      },
      {
         "Effect":"Allow",
         "Action":[
            "ssm:SendCommand"
         ],
         "Resource":[
            "arn:aws:ssm:us-west-1::document/AWS-*",
            "arn:aws:ssm:us-east-2::document/AWS-*"
         ]
      },
      {
```

```
            "Effect":"Allow",
            "Action":[
               "ssm:UpdateInstanceInformation",
               "ssm:ListCommands",
               "ssm:ListCommandInvocations",
               "ssm:GetDocument"
            ],
            "Resource":"*"
         }
      ]
}
```

Vous pouvez également créer des stratégies IAM qui permettent à un utilisateur d'exécuter des commandes sur plusieurs groupes d'instances balisées. La stratégie suivante permet à l'utilisateur d'exécuter des commandes sur un des deux groupes d'instances balisées, ou les deux.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "ssm:SendCommand"
         ],
         "Resource":"*",
         "Condition":{
            "StringLike":{
               "ssm:resourceTag/tag_key1":[
                  "tag_value1"
               ]
            }
         }
      },
      {
         "Effect":"Allow",
         "Action":[
            "ssm:SendCommand"
         ],
         "Resource":"*",
         "Condition":{
            "StringLike":{
               "ssm:resourceTag/tag_key2":[
                  "tag_value2"
               ]
            }
         }
      },
      {
         "Effect":"Allow",
         "Action":[
            "ssm:SendCommand"
         ],
         "Resource":[
            "arn:aws:ssm:us-west-1::document/AWS-*",
            "arn:aws:ssm:us-east-2::document/AWS-*"
         ]
      },
      {
         "Effect":"Allow",
         "Action":[
            "ssm:UpdateInstanceInformation",
            "ssm:ListCommands",
            "ssm:ListCommandInvocations",
            "ssm:GetDocument"
```

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

Pour de plus amples informations sur la création des stratégies utilisateur IAM, veuillez consulter Stratégies gérées et stratégies en ligne dans le IAM Guide de l'utilisateur. Pour de plus amples informations sur le balisage des instances, veuillez consulter Balisage de vos ressources Amazon EC2 dans le Amazon EC2 Guide de l'utilisateur pour les instances Linux (le contenu s'applique aux instances Windows et Linux).

Exécuter des commandes avec Systems Manager Fonctionnalité Exécuter la commande

Cette section inclut des informations sur l'envoi de commandes à partir de la console AWS Systems Manager et sur l'envoi de commandes à un parc d'instances à l'aide du paramètre `Targets` avec des balises EC2. Cette section inclut également des informations sur comment annuler une commande.

> **Tip**
>
> Si vous utilisez la fonctionnalité Fonctionnalité Exécuter la commande pour la première fois, nous vous recommandons d'exécuter les commandes sur une instance de test ou sur une instance qui n'est pas utilisée dans un environnement de production.

Pour plus d'informations sur l'envoi de commandes à l'aide PowerShell de Windows, consultez Procédure : Utilisation de l' Outils AWS pour Windows PowerShell avecFonctionnalité Exécuter la commande (p. 977) ou les exemples dans le AWS Systems Manager section of the Outils AWS pour PowerShell Cmdlet Reference . Pour de plus amples informations sur l'envoi de commandes à l'aide de l'interface de ligne de commande (CLI) AWS, veuillez consulter Procédure : Utilisation de l' AWS CLI avecFonctionnalité Exécuter la commande (p. 971) ou la référence SSM.

> **Important**
>
> Lorsque vous exécutez une commande dans Fonctionnalité Exécuter la commande, n'incluez aucune information sensible formatée en texte brut, comme les mots de passe, les données de configuration ou d'autres secrets. L'ensemble de l'activité de l'API Systems Manager de votre compte est enregistrée dans un compartiment Amazon S3, dans des journaux AWS CloudTrail. Cela signifie que tout utilisateur ayant accès à ce compartiment S3 peut afficher les valeurs en texte brut de ces secrets. Pour cette raison, nous vous recommandons fortement de créer et d'utiliser des paramètres `SecureString` pour chiffrer les données sensibles que vous utilisez dans vos opérations Systems Manager.
> Pour plus d'informations, consultez la section Restriction de l'accès aux paramètres Systems Manager à l'aide des stratégies IAM (p. 223).

Table des matières

- Exécution des commandes depuis la console (p. 946)
- Exécution PowerShell de scripts sur des instances Linux (p. 952)
- Exécution de commandes à l'aide du paramètre de version de document (p. 953)
- Utilisation des contrôles de cibles et de taux pour envoyer des commandes à un parc (p. 955)
- Annulation d'une commande (p. 962)

Exécution des commandes depuis la console

Vous pouvez utiliser Fonctionnalité Exécuter la commande pour configurer des instances depuis la console sans avoir à vous connecter à chacune d'entre elles. Cette rubrique comprend un exemple qui montre

comment mettre à jour Agent SSM (p. 948) sur une instance à l'aide de la fonctionnalité Fonctionnalité Exécuter la commande.

Avant de commencer

Avant d'envoyer une commande à l'aide de la fonctionnalité Fonctionnalité Exécuter la commande, vérifiez que vos instances répondent aux exigences (p. 15) de Systems Manager.

Pour envoyer une commande à l'aide de la fonctionnalité Fonctionnalité Exécuter la commande

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fonctionnalité Exécuter la commande.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Fonctionnalité Exécuter la commande.
3. Choisissez Exécuter une commande.
4. Dans la liste Command document (Document de commande), choisissez un document Systems Manager.
5. Dans la section Command parameters (Paramètres de la commande), indiquez des valeurs pour les paramètres requis.
6. Dans la section Cibles, identifiez les instances sur lesquelles vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant manuellement les instances ou en indiquant un groupe de ressources.

   > **Note**
   >
   > Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.
7. Pour Autres paramètres :
   - Pour Comment (Commentaire), saisissez des informations à propos de cette commande.
   - Pour Délai (secondes), précisez le nombre de secondes durant lesquelles le système doit attendre avant de mettre en échec l'exécution de la commande globale.
8. (Facultatif) Pour Contrôle du débit :
   - Pour Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage d'instances sur lesquelles exécuter simultanément la commande.

     > **Note**
     >
     > Si vous avez sélectionné des cibles en indiquant des balises appliquées à des instances gérées ou en indiquant des groupes de ressources AWS, et que vous n'êtes pas sûr du nombre d'instances ciblées, limitez le nombre d'instances pouvant exécuter simultanément le document en spécifiant un pourcentage.
   - Pour Seuil d'erreur, spécifiez quand arrêter l'exécution de la commande sur d'autres instances après son échec sur un nombre ou un pourcentage d'instances. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les instances sur lesquelles la commande est en cours de traitement peuvent également envoyer des erreurs.
9. (Facultatif) Pour Output options (Options de sortie), pour enregistrer la sortie de la commande dans un fichier, cochez la case Write command output to an S3 bucket (Écrire la sortie de commande vers un compartiment S3). Entrez les noms de compartiment et de préfixe (dossier) dans les zones.

Note

Les autorisations S3 qui accordent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour de plus amples informations, veuillez consulter Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). En outre, si le compartiment S3 spécifié se trouve dans un compte AWS différent, assurez-vous que le profil d'instance associé à l'instance dispose des autorisations nécessaires pour écrire dans ce compartiment.

10. Dans la section SNS notifications (Notifications SNS), si vous souhaitez que des notifications sur le statut de l'exécution de la commande soient envoyées, cochez la case Enable SNS notifications (Activer les notifications SNS).

    Pour plus d'informations sur la configuration des notifications Amazon SNS pour la fonctionnalité Exécuter la commande, consultez Surveillance des changements de statut Systems Manager à l'aide de notifications Amazon SNS (p. 1344).
11. Cliquez sur Exécuter.

Pour plus d'informations sur l'annulation d'une commande, consultez the section called “Annulation d'une commande” (p. 962).

Réexécution des commandes

Systems Manager comprend deux options pour vous aider à réexécuter une commande à partir de la page Exécuter la commande de la console AWS Systems Manager.

- Réexécuter : Ce bouton vous permet d'exécuter la même commande sans y apporter de modifications.
- Copier vers le nouveau : Ce bouton copie les paramètres d'une commande dans une nouvelle commande et vous donne la possibilité de modifier ces paramètres avant de l'exécuter.

Pour réexécuter une commande

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fonctionnalité Exécuter la commande.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Fonctionnalité Exécuter la commande.
3. Choisissez une commande à réexécuter. Vous pouvez réexécuter une commande immédiatement après l'avoir exécutée à partir de la page de détails de la commande. Vous pouvez également choisir une commande que vous avez précédemment exécutée dans l'onglet Historique des commandes.
4. Choisissez Réexécuter pour exécuter la même commande sans modifications, ou choisissez Copier vers nouveau pour modifier les paramètres de la commande avant de l'exécuter.

Mettre à jour l'Agent SSM à l'aide de Fonctionnalité Exécuter la commande

La procédure suivante décrit comment rapidement mettre à jour l'Agent SSM en cours d'exécution sur vos instances Windows Server et Linux. Vous pouvez mettre à jour à l'aide de la dernière version ou d'une version plus ancienne. Lorsque vous exécutez la commande, le système télécharge la version d'AWS, l'installe, puis désinstalle la version qui existait avant l'exécution de la commande. Si une erreur se produit au cours de ce processus, le système revient à la version du serveur avant l'exécution de la commande et le statut de cette dernière indique qu'elle a échoué.

Note

Notez les détails suivants sur la mise à jour automatique Agent SSM d' :

- À compter du 21 septembre 2020, la mise à jour automatique installe Agent SSM version 3.0. Pour de plus amples informations, veuillez consulter Agent SSMversion 3 (p. 75).
- Pour recevoir des notifications sur les Agent SSM mises à jour, abonnez-vous à la page Agent SSM Notes de mise à jour sur GitHub .

Pour mettre à jour l'Agent SSM en utilisant Fonctionnalité Exécuter la commande

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fonctionnalité Exécuter la commande.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Fonctionnalité Exécuter la commande.
3. Choisissez Exécuter une commande.
4. Dans la liste Command document, choisissez AWS-UpdateSSMAgent.
5. Dans la section Paramètres de la commande, indiquez des valeurs pour les paramètres suivants, si vous le souhaitez :

   a. (Facultatif) Pour Version, tapez la version de l'Agent SSM à installer. Vous pouvez installer des versions plus anciennes de l'agent. Si vous ne spécifiez pas de version, le service installe la dernière version.

   b. (Facultatif) Pour Allow Downgrade (Autoriser le retour à la version précédente), choisissez true pour installer une version antérieure de l'Agent SSM. Si vous sélectionnez cette option, vous devez spécifier le numéro de version antérieure. Choisissez false pour installer uniquement la dernière version du service.
6. Dans la section Cibles, identifiez les instances sur lesquelles vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant manuellement les instances ou en indiquant un groupe de ressources.

   Note

   Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.
7. Pour Autres paramètres :

   - Pour Comment (Commentaire), saisissez des informations à propos de cette commande.
   - Pour Délai (secondes), précisez le nombre de secondes durant lesquelles le système doit attendre avant de mettre en échec l'exécution de la commande globale.
8. (Facultatif) Pour Contrôle du débit :

   - Pour Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage d'instances sur lesquelles exécuter simultanément la commande.

     Note

     Si vous avez sélectionné des cibles en indiquant des balises appliquées à des instances gérées ou en indiquant des groupes de ressources AWS, et que vous n'êtes pas sûr du nombre d'instances ciblées, limitez le nombre d'instances pouvant exécuter simultanément le document en spécifiant un pourcentage.

- Pour Seuil d'erreur, spécifiez quand arrêter l'exécution de la commande sur d'autres instances après son échec sur un nombre ou un pourcentage d'instances. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les instances sur lesquelles la commande est en cours de traitement peuvent également envoyer des erreurs.

9. (Facultatif) Pour Output options (Options de sortie), pour enregistrer la sortie de la commande dans un fichier, cochez la case Write command output to an S3 bucket (Écrire la sortie de commande vers un compartiment S3). Entrez les noms de compartiment et de préfixe (dossier) dans les zones.

   Note

   Les autorisations S3 qui accordent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour de plus amples informations, veuillez consulter Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). En outre, si le compartiment S3 spécifié se trouve dans un compte AWS différent, assurez-vous que le profil d'instance associé à l'instance dispose des autorisations nécessaires pour écrire dans ce compartiment.

10. Dans la section SNS notifications (Notifications SNS), si vous souhaitez que des notifications sur le statut de l'exécution de la commande soient envoyées, cochez la case Enable SNS notifications (Activer les notifications SNS).

    Pour plus d'informations sur la configuration des notifications Amazon SNS pour la fonctionnalité Exécuter la commande, consultez Surveillance des changements de statut Systems Manager à l'aide de notifications Amazon SNS (p. 1344).

11. Cliquez sur Exécuter.

Mise à jour à PowerShell l'aide d'Fonctionnalité Exécuter la commande

La procédure suivante décrit comment mettre à jour PowerShell vers la version 5.1 sur vos instances Windows Server 2012 et 2012 R2. Le script fourni dans cette procédure télécharge la mise à jour Windows Management Framework (AnalyzeF) version 5.1 et démarre l'installation de la mise à jour. L'instance redémarre au cours de ce processus, car cela est nécessaire lors de l'installation de WMF 5.1. Le téléchargement et l'installation de la mise à jour durent environ cinq minutes.

Pour mettre à jour l'PowerShell en utilisant Fonctionnalité Exécuter la commande

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fonctionnalité Exécuter la commande.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Fonctionnalité Exécuter la commande.
3. Choisissez Exécuter une commande.
4. Dans la liste Document de commande, choisissez AWS-RunPowerShellScript.
5. Dans la section Commands, collez les commandes suivantes pour votre système d'exploitation.

   Windows Server 2012 R2

```
Set-Location -Path "C:\Windows\Temp"

Invoke-WebRequest "https://go.microsoft.com/fwlink/?linkid=839516" -OutFile
 "Win8.1AndW2K12R2-KB3191564-x64.msu"
```

```
Start-Process -FilePath "$env:systemroot\system32\wusa.exe" -Verb RunAs -
ArgumentList ('Win8.1AndW2K12R2-KB3191564-x64.msu', '/quiet')
```

Windows Server 2012

```
Set-Location -Path "C:\Windows\Temp"

Invoke-WebRequest "https://go.microsoft.com/fwlink/?linkid=839513" -OutFile "W2K12-
KB3191565-x64.msu"

Start-Process -FilePath "$env:systemroot\system32\wusa.exe" -Verb RunAs -
ArgumentList ('W2K12-KB3191565-x64.msu', '/quiet')
```

6. Dans la section Cibles, identifiez les instances sur lesquelles vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant manuellement les instances ou en indiquant un groupe de ressources.

   Note

   Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.

7. Pour Autres paramètres :

   - Pour Comment (Commentaire), saisissez des informations à propos de cette commande.
   - Pour Délai (secondes), précisez le nombre de secondes durant lesquelles le système doit attendre avant de mettre en échec l'exécution de la commande globale.

8. (Facultatif) Pour Contrôle du débit :

   - Pour Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage d'instances sur lesquelles exécuter simultanément la commande.

     Note

     Si vous avez sélectionné des cibles en indiquant des balises appliquées à des instances gérées ou en indiquant des groupes de ressources AWS, et que vous n'êtes pas sûr du nombre d'instances ciblées, limitez le nombre d'instances pouvant exécuter simultanément le document en spécifiant un pourcentage.

   - Pour Seuil d'erreur, spécifiez quand arrêter l'exécution de la commande sur d'autres instances après son échec sur un nombre ou un pourcentage d'instances. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les instances sur lesquelles la commande est en cours de traitement peuvent également envoyer des erreurs.

9. (Facultatif) Pour Output options (Options de sortie), pour enregistrer la sortie de la commande dans un fichier, cochez la case Write command output to an S3 bucket (Écrire la sortie de commande vers un compartiment S3). Entrez les noms de compartiment et de préfixe (dossier) dans les zones.

   Note

   Les autorisations S3 qui accordent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour de plus amples informations, veuillez consulter Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). En outre, si le compartiment S3 spécifié se trouve dans un compte AWS différent, assurez-vous que le profil d'instance associé à l'instance dispose des autorisations nécessaires pour écrire dans ce compartiment.

10. Dans la section SNS notifications (Notifications SNS), si vous souhaitez que des notifications sur le statut de l'exécution de la commande soient envoyées, cochez la case Enable SNS notifications (Activer les notifications SNS).

Pour plus d'informations sur la configuration des notifications Amazon SNS pour la fonctionnalité Exécuter la commande, consultez Surveillance des changements de statut Systems Manager à l'aide de notifications Amazon SNS (p. 1344).

11. Cliquez sur Exécuter.

Une fois que l'instance a redémarré et que l'installation de la mise à jour est terminée, connectez-vous à votre instance pour confirmer que la mise à niveau vers la version 5.1 PowerShell a réussi. Pour vérifier la version d' PowerShell sur votre instance, ouvrez PowerShell et entrez `$PSVersionTable` . La `PSVersion` valeur dans le tableau de sortie indique 5.1 si la mise à niveau a réussi.

Si la `PSVersion` valeur est différente de 5.1, par exemple 3.0 ou 4.0, passez en revue les journaux Setup dans l'Observateur d'événements sous Journaux Windows. Ces journaux indiquent pourquoi l'installation de la mise à jour a échoué.

Exécution PowerShell de scripts sur des instances Linux

Avec le `aws:runPowerShellScript` plug-in ou le document `AWS-RunPowerShellScript` de commande, ainsi que PowerShell Core, vous pouvez exécuter PowerShell des scripts sur les instances Linux. Cela peut être utile pour les administrateurs système qui connaissent PowerShell et qui le préfèrent à d'autres langages de script.

Avant de commencer

Connectez-vous à votre instance Linux et suivez la procédure d'installation PowerShell de Core pour le système d'exploitation approprié.

> **Note**
>
> De nombreuses PowerShell commandes (applets de commande) ne sont pas disponibles sous Linux. Pour voir quelles commandes sont disponibles, utilisez l'applet de commande `Get-Command` après avoir démarré PowerShell à l'aide de la commande `pwsh` sur votre instance Linux. Pour de plus amples informations, veuillez consulter Get-Command.

La procédure suivante décrit comment exécuter un PowerShell script sur une instance Linux à l'aide de la console.

Pour exécuter un PowerShell script sur une instance Linux à l'aide de la console

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fonctionnalité Exécuter la commande.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Fonctionnalité Exécuter la commande.
3. Choisissez Exécuter une commande.
4. Dans la liste Document de commande, choisissez un document `AWS-RunPowerShellScript`.
5. Dans la section Paramètres de commande, spécifiez les PowerShell commandes disponibles que vous souhaitez utiliser.
6. Dans la section Cibles, identifiez les instances sur lesquelles vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant manuellement les instances ou en indiquant un groupe de ressources.

Note

Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.

7. Pour Autres paramètres :
   - Pour Comment (Commentaire), saisissez des informations à propos de cette commande.
   - Pour Délai (secondes), précisez le nombre de secondes durant lesquelles le système doit attendre avant de mettre en échec l'exécution de la commande globale.
8. (Facultatif) Pour Contrôle du débit :
   - Pour Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage d'instances sur lesquelles exécuter simultanément la commande.

     Note

     Si vous avez sélectionné des cibles en indiquant des balises appliquées à des instances gérées ou en indiquant des groupes de ressources AWS, et que vous n'êtes pas sûr du nombre d'instances ciblées, limitez le nombre d'instances pouvant exécuter simultanément le document en spécifiant un pourcentage.
   - Pour Seuil d'erreur, spécifiez quand arrêter l'exécution de la commande sur d'autres instances après son échec sur un nombre ou un pourcentage d'instances. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les instances sur lesquelles la commande est en cours de traitement peuvent également envoyer des erreurs.
9. (Facultatif) Pour Output options (Options de sortie), pour enregistrer la sortie de la commande dans un fichier, cochez la case Write command output to an S3 bucket (Écrire la sortie de commande vers un compartiment S3). Entrez les noms de compartiment et de préfixe (dossier) dans les zones.

   Note

   Les autorisations S3 qui accordent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour de plus amples informations, veuillez consulter Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). En outre, si le compartiment S3 spécifié se trouve dans un compte AWS différent, assurez-vous que le profil d'instance associé à l'instance dispose des autorisations nécessaires pour écrire dans ce compartiment.
10. Dans la section SNS notifications (Notifications SNS), si vous souhaitez que des notifications sur le statut de l'exécution de la commande soient envoyées, cochez la case Enable SNS notifications (Activer les notifications SNS).

    Pour plus d'informations sur la configuration des notifications Amazon SNS pour la fonctionnalité Exécuter la commande, consultez Surveillance des changements de statut Systems Manager à l'aide de notifications Amazon SNS (p. 1344).
11. Cliquez sur Exécuter.

Pour voir les exemples qui utilisent le plug-in `aws:runPowerShellScript`, veuillez consulter aws:runPowerShellScript (p. 1257).

Exécution de commandes à l'aide du paramètre de version de document

Vous pouvez utiliser le paramètre document-version pour spécifier la version d'un document SSM à utiliser lors de l'exécution de la commande. Vous pouvez spécifier l'une des options suivantes pour ce paramètre :

- $DEFAULT
- $LATEST
- Numéro de version

Utilisez la procédure suivante pour exécuter une commande à l'aide du paramètre document-version.

Linux

Pour exécuter des commandes à l'aide de AWS CLI sur des machines Linux locales

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).
2. Répertorier tous les documents disponibles

   Cette commande répertorie tous les documents disponibles pour votre compte en fonction des autorisations IAM.

```
aws ssm list-documents
```

3. Utilisez la commande suivante pour afficher les différentes versions d'un document.

```
aws ssm list-document-versions \
    --name "document-name"
```

4. Utilisez la commande suivante pour exécuter une commande qui utilise une version de document SSM.

```
aws ssm send-command \
    --document-name "AWS-RunShellScript" \
    --parameters commands="echo Hello" \
    --instance-ids instance-ID \
    --document-version '$LATEST'
```

Windows

Pour exécuter des commandes à l'aide de AWS CLI sur des machines Windows locales

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).
2. Répertorier tous les documents disponibles

   Cette commande répertorie tous les documents disponibles pour votre compte en fonction des autorisations IAM.

```
aws ssm list-documents
```

3. Utilisez la commande suivante pour afficher les différentes versions d'un document.

```
aws ssm list-document-versions ^
    --name "document-name"
```

4. Utilisez la commande suivante pour exécuter une commande qui utilise une version de document SSM.

```
aws ssm send-command ^
    --document-name "AWS-RunShellScript" ^
    --parameters commands="echo Hello" ^
    --instance-ids instance-ID ^
    --document-version "$LATEST"
```

PowerShell

Pour exécuter des commandes à l'aide de l'Outils pour PowerShell

1. Si vous ne l'avez pas déjà fait, installez et configurez l'Outils AWS pour PowerShell.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Répertorier tous les documents disponibles

   Cette commande répertorie tous les documents disponibles pour votre compte en fonction des autorisations IAM. Elle renvoie une liste des documents Linux et Windows.

```
Get-SSMDocumentList
```

3. Utilisez la commande suivante pour afficher les différentes versions d'un document.

```
Get-SSMDocumentVersionList `
    -Name "document-name"
```

4. Utilisez la commande suivante pour exécuter une commande qui utilise une version de document SSM.

```
Send-SSMCommand `
    -DocumentName "AWS-RunShellScript" `
    -Parameter @{commands = "echo helloWorld"} `
    -InstanceIds "instance-ID" `
    -DocumentVersion $LATEST
```

Utilisation des contrôles de cibles et de taux pour envoyer des commandes à un parc

Vous pouvez envoyer des commandes à des dizaines, des centaines ou des milliers d'instances à l'aide du paramètre `targets` (option Spécifier les balises d'instance de la page Exécuter une commande de la console). Le paramètre `targets` accepte une combinaison `Key,Value` basée sur les balises Amazon EC2 que vous avez spécifiées pour vos instances. Lorsque vous exécutez la commande, le système recherche les fichiers et tente d'exécuter la commande sur toutes les instances qui correspondent aux balises spécifiées. Pour de plus amples informations sur les balises Amazon EC2, veuillez consulter Balisage de vos ressources Amazon EC2 dans le Guide de l'utilisateur Amazon EC2 (le contenu s'applique aux instances Windows Server et Linux). Vous pouvez également envoyer des commandes à des instances qui appartiennent à un groupe de ressources AWS. Pour plus d'informations sur les groupes de ressources, consultez Que sont les groupes de ressources ? dans le Guide de l'utilisateur Groupes de ressources AWS.

> **Note**
>
> Vous pouvez également utiliser le `targets` paramètre pour cibler une liste d'instances spécifiques, comme décrit dans la section suivante.IDs

Pour contrôler l'exécution d'une commande sur des centaines de milliers d'instances, la fonctionnalité Fonctionnalité Exécuter la commande inclut également des paramètres pour limiter le nombre d'instances pouvant traiter simultanément une demande et le nombre d'erreurs pouvant être émises par une commande avant que la commande ne prenne fin.

Table des matières

- Ciblage de plusieurs Instances (p. 956)
- Utilisation des contrôles de taux (p. 960)

Ciblage de plusieurs Instances

Vous pouvez exécuter une commande et cibler des instances en spécifiant des balises appliquées à des instances gérées, des noms de groupes de ressources AWS ou des instances IDs .

> **Note**
>
> Les exemples de commandes de cette section sont tronqués à l'aide de `[...]`.

Pour une utilisation avec la commande `send-command` de l'AWS CLI, le paramètre `targets` prend en charge la syntaxe illustrée dans les exemples suivants :

Exemple 1 : Ciblage des balises

Linux

```
aws ssm send-command \
 --document-name document-name \
 --targets Key=tag:tag-name,Values=tag-value \
 [...]
```

Windows

```
aws ssm send-command ^
 --document-name document-name ^
 --targets Key=tag:tag-name,Values=tag-value ^
 [...]
```

Exemple 2 : Ciblage d'un groupe de ressources AWS par nom

Vous pouvez spécifier au maximum un nom de groupe de ressources par commande. Lorsque vous créez un groupe de ressources, nous vous recommandons d'inclure `AWS::SSM:ManagedInstance` et `AWS::EC2::Instance` comme types de ressource dans vos critères de regroupement.

> **Note**
>
> Pour pouvoir envoyer des commandes qui ciblent un groupe de ressources, vous devez avoir reçu des autorisations IAM pour répertorier, ou afficher, les ressources qui appartiennent à ce groupe. Pour plus d'informations, consultez Configuration d'autorisations dans le Guide de l'utilisateur Groupes de ressources AWS.

Linux

```
aws ssm send-command \
 --document-name document-name \
```

```
 --targets Key=resource-groups:Name,Values=resource-group-name \
 [...]
```

Windows

```
aws ssm send-command ^
 --document-name document-name ^
 --targets Key=resource-groups:Name,Values=resource-group-name ^
 [...]
```

Exemple 3 : Ciblage d'un groupe de ressources AWS par type de ressource

Vous pouvez spécifier un maximum de cinq types de groupes de ressources par commande. Lorsque vous créez un groupe de ressources, nous vous recommandons d'inclure `AWS::SSM:ManagedInstance` et `AWS::EC2::Instance` comme types de ressource dans vos critères de regroupement.

> **Note**
>
> Pour pouvoir envoyer des commandes qui ciblent un groupe de ressources, vous devez avoir reçu des autorisations IAM pour répertorier, ou afficher, les ressources qui appartiennent à ce groupe. Pour plus d'informations, consultez Configuration d'autorisations dans le Guide de l'utilisateur Groupes de ressources AWS.

Linux

```
aws ssm send-command \
 --document-name document-name \
 --targets Key=resource-groups:ResourceTypeFilters,Values=resource-type-1,resource-
type-2 \
 [...]
```

Windows

```
aws ssm send-command ^
 --document-name document-name ^
 --targets Key=resource-groups:ResourceTypeFilters,Values=resource-type-1,resource-
type-2 ^
 [...]
```

Exemple 4 : Ciblage de l'instanceIDs

Linux

```
aws ssm send-command \
 --document-name document-name \
 --targets Key=instanceids,Values=instance-ID-1,instance-ID-2,instance-ID-3 \
 [...]
```

Windows

```
aws ssm send-command ^
 --document-name document-name ^
 --targets Key=instanceids,Values=instance-ID-1,instance-ID-2,instance-ID-3 ^
 [...]
```

Si vous avez balisé des instances pour différents environnements à l'aide d'une balise `Key` nommée `Environment` et de valeurs `Values` `Development`, `Test`, `Pre-production` et `Production`, vous

pouvez envoyer une commande à toutes les instances dans l'un de ces environnements à l'aide du paramètre avec la syntaxe suivante :`targets`

Linux

```
aws ssm send-command \
 --document-name document-name \
 --targets Key=tag:Environment,Values=Development \
 [...]
```

Windows

```
aws ssm send-command ^
 --document-name document-name ^
 --targets Key=tag:Environment,Values=Development ^
 [...]
```

Vous pouvez cibler des instances supplémentaires dans d'autres environnements en ajoutant un élément à la liste `Values`. Séparez les éléments avec des virgules.

Linux

```
aws ssm send-command \
 --document-name document-name \
 --targets Key=tag:Environment,Values=Development,Test,Pre-production \
 [...]
```

Windows

```
aws ssm send-command ^
 --document-name document-name ^
 --targets Key=tag:Environment,Values=Development,Test,Pre-production ^
 [...]
```

Variation : Ajustement de vos cibles à l'aide de plusieurs `Key` critères

Vous pouvez affiner le nombre de cibles pour votre commande en incluant plusieurs critères `Key`. Si vous incluez plusieurs critères `Key`, le système cible les instances qui répondent à tous les critères. La commande suivante cible toutes les instances balisées pour le service financier et balisées pour le rôle de serveur de base de données.

Linux

```
aws ssm send-command \
 --document-name document-name \
 --targets Key=tag:Department,Values=Finance Key=tag:ServerRole,Values=Database \
 [...]
```

Windows

```
aws ssm send-command ^
 --document-name document-name ^
 --targets Key=tag:Department,Values=Finance Key=tag:ServerRole,Values=Database ^
 [...]
```

Variation : Utilisation de plusieurs `Key` critères`Value`

En reprenant l'exemple précédent, vous pouvez cibler plusieurs services et plusieurs rôles de serveur en incluant des éléments supplémentaires dans les critères `Values`.

Linux

```
aws ssm send-command \
 --document-name document-name \
 --targets Key=tag:Department,Values=Finance,Marketing
 Key=tag:ServerRole,Values=WebServer,Database \
 [...]
```

Windows

```
aws ssm send-command ^
 --document-name document-name ^
 --targets Key=tag:Department,Values=Finance,Marketing
 Key=tag:ServerRole,Values=WebServer,Database ^
 [...]
```

Variation : Ciblage des instances balisées à l'aide de plusieurs `Values` critères

Si vous avez balisé des instances pour différents environnements à l'aide d'une balise `Key` nommée `Department` et de valeurs `Values`, `Sales` et `Finance`, vous pouvez envoyer une commande à toutes les instances dans l'un de ces environnements à l'aide du paramètre `targets` avec la syntaxe suivante :

Linux

```
aws ssm send-command \
 --document-name document-name \
 --targets Key=tag:Department,Values=Sales,Finance \
 [...]
```

Windows

```
aws ssm send-command ^
 --document-name document-name ^
 --targets Key=tag:Department,Values=Sales,Finance ^
 [...]
```

> **Note**
>
> Vous pouvez spécifier un maximum de cinq clés et cinq valeurs pour chaque clé.

Si une clé de balise (le nom de la balise) ou une valeur de balise inclut des espaces, vous devez placer la clé ou la valeur de balise entre guillemets, comme illustré dans les exemples suivants.

Exemple: Espaces dans la `Value` balise

Linux

```
aws ssm send-command \
 --document-name document-name \
 --targets Key=tag:OS,Values="Windows Server 2016 Nano" \
 [...]
```

Windows

```
aws ssm send-command ^
```

```
 --document-name document-name ^
 --targets Key=tag:OS,Values="Windows Server 2016 Nano" ^
 [...]
```

Exemple: Des espaces dans `tag` la clé et`Value`

Linux

```
aws ssm send-command \
 --document-name document-name \
 --targets Key="tag:Operating System",Values="Windows Server 2016 Nano" \
 [...]
```

Windows

```
aws ssm send-command ^
 --document-name document-name ^
 --targets Key="tag:Operating System",Values="Windows Server 2016 Nano" ^
 [...]
```

Exemple: Des espaces dans un élément d'une liste de`Values`

Linux

```
aws ssm send-command \
 --document-name document-name \
 --targets Key=tag:Department,Values="Sales","Finance","Systems Mgmt" \
 [...]
```

Windows

```
aws ssm send-command ^
 --document-name document-name ^
 --targets Key=tag:Department,Values="Sales","Finance","Systems Mgmt" ^
 [...]
```

Utilisation des contrôles de taux

Vous pouvez contrôler le taux d'envoi des commandes aux instances d'un groupe à l'aide des contrôles de concurrence et des contrôles d'erreur.

Rubriques

- Utilisation de contrôles d'accès simultanés (p. 960)
- Utilisation de contrôles d'erreur (p. 961)

Utilisation de contrôles d'accès simultanés

Vous pouvez contrôler le nombre de serveurs exécutant la commande en même temps à l'aide du paramètre `max-concurrency` (options Concurrency de la page Exécuter une commande). Vous pouvez spécifier un nombre absolu d'instances, par exemple 10, ou un pourcentage de l'ensemble de la cible, par exemple 10 %. Le système de mise en file d'attente transmet la commande à une seule instance et attend que l'appel initial soit confirmé avant d'envoyer la commande à deux autres instances. Le système envoie de façon exponentielle des commandes à plusieurs instances jusqu'à ce que la valeur `max-concurrency`

soit atteinte. La valeur par défaut de `max-concurrency` est 50. Les exemples suivants vous montrent comment spécifier des valeurs pour le paramètre `max-concurrency` :

Linux

```
aws ssm send-command \
 --document-name document-name \
 --max-concurrency 10 \
 --targets Key=tag:Environment,Values=Development \
 [...]
```

```
aws ssm send-command \
 --document-name document-name \
 --max-concurrency 10% \
 --targets Key=tag:Department,Values=Finance,Marketing
 Key=tag:ServerRole,Values=WebServer,Database \
 [...]
```

Windows

```
aws ssm send-command ^
 --document-name document-name ^
 --max-concurrency 10 ^
 --targets Key=tag:Environment,Values=Development ^
 [...]
```

```
aws ssm send-command ^
 --document-name document-name ^
 --max-concurrency 10% ^
 --targets Key=tag:Department,Values=Finance,Marketing
 Key=tag:ServerRole,Values=WebServer,Database ^
 [...]
```

Utilisation de contrôles d'erreur

Vous pouvez également contrôler l'exécution d'une commande sur des centaines ou des milliers d'instances en définissant une limite d'erreurs à l'aide des paramètres `max-errors` (champ Error threshold (Seuil d'erreur) de la page Exécuter une commande). Le paramètre spécifie le nombre d'erreurs autorisées avant que le système cesse d'envoyer la commande d'autres instances. Vous pouvez spécifier un nombre absolu d'erreurs, par exemple **`10`** , ou un pourcentage de l'ensemble de la cible, par exemple **`10%`** . Par exemple, si vous spécifiez **`3`** , le système cesse d'envoyer la commande à la réception de la quatrième erreur. Si vous spécifiez **`0`**, le système cesse d'envoyer la commande à des instances supplémentaires une fois que le premier résultat d'erreur est renvoyé. Si vous envoyez une commande à 50 instances et que vous la définissez sur `max-errors` , le système cesse d'envoyer la commande à des instances supplémentaires à la réception de la sixième erreur.**`10%`**

Les appels qui exécutent déjà une commande lorsque `max-errors` est atteint sont autorisés à se terminer, mais certains de ces appels peuvent également échouer. Si vous devez vous assurer que le nombre d'appels ayant échoué ne dépassera pas la valeur de `max-errors`, définissez `max-concurrency` sur **`1`** pour que les appels soit exécutés un par un. La valeur par défaut pour max-errors est 0. Les exemples suivants vous montrent comment spécifier des valeurs pour le paramètre `max-errors` :

Linux

```
aws ssm send-command \
 --document-name document-name \
```

```
 --max-errors 10 \
 --targets Key=tag:Database,Values=Development \
 [...]
```

```
aws ssm send-command \
 --document-name document-name \
 --max-errors 10% \
 --targets Key=tag:Environment,Values=Development \
 [...]
```

```
aws ssm send-command \
 --document-name document-name \
 --max-concurrency 1 \
 --max-errors 1 \
 --targets Key=tag:Environment,Values=Production \
 [...]
```

Windows

```
aws ssm send-command ^
 --document-name document-name ^
 --max-errors 10 ^
 --targets Key=tag:Database,Values=Development ^
 [...]
```

```
aws ssm send-command ^
 --document-name document-name ^
 --max-errors 10% ^
 --targets Key=tag:Environment,Values=Development ^
 [...]
```

```
aws ssm send-command ^
 --document-name document-name ^
 --max-concurrency 1 ^
 --max-errors 1 ^
 --targets Key=tag:Environment,Values=Production ^
 [...]
```

Annulation d'une commande

Vous pouvez tenter d'annuler une commande tant que le service est associé à l'état En attente ou En cours d'exécution. Toutefois, même si une commande est encore associée à l'un de ces états, nous ne pouvons pas garantir qu'elle sera mise hors service et que le processus sous-jacent sera arrêté.

Pour annuler une commande via la console

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fonctionnalité Exécuter la commande.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Fonctionnalité Exécuter la commande.
3. Sélectionnez l'appel de commande que vous souhaitez annuler.
4. Choisissez Annuler la commande.

Pour annuler une commande à l'aide de l'AWS CLI

Utilisez la commande suivante :

Linux

```
aws ssm cancel-command \
 --command-id "command-ID" \
 --instance-ids "instance-ID"
```

Windows

```
aws ssm cancel-command ^
 --command-id "command-ID" ^
 --instance-ids "instance-ID"
```

Pour de plus amples informations sur le statut d'une commande annulée, veuillez consulter Comprendre les états des commandes (p. 965).

Gestion des codes de sortie avec des scripts

Dans certains cas, vous devrez peut-être gérer la façon dont les codes de sortie sont gérés dans vos commandes à l'aide de scripts.

Rubriques

- Redémarrage d'une instance gérée à partir de scripts (p. 963)
- Gestion des codes de sortie dans les commandes Fonctionnalité Exécuter la commande (p. 964)

Redémarrage d'une instance gérée à partir de scripts

Si les scripts que vous exécutez à l'aide de la fonctionnalité Fonctionnalité Exécuter la commande redémarrent les instances gérées, vous devez spécifier un code de sortie dans votre script. Si vous essayez de redémarrer une instance à partir d'un script à l'aide d'un autre mécanisme, l'état d'exécution des scripts peut ne pas être mis à jour correctement, même si le redémarrage est la dernière étape dans votre script. Pour les instances Windows gérées, spécifiez `exit 3010` dans votre script. Pour les instances Linux gérées, vous spécifiez `exit 194` . Le code de sortie indique Agent SSM à de redémarrer l'instance gérée, puis de redémarrer le script une fois le redémarrage terminé. Avant de commencer le redémarrage, l'Agent SSM informe le service Systems Manager dans le cloud que les communications seront interrompues pendant le redémarrage du serveur.

Création de scripts idempotents

Lorsque vous développez des scripts qui redémarrent les instances gérées, rendez les scripts idempotents de sorte que l'exécution du script continue là où elle s'était arrêtée après le redémarrage. Les scripts idempotents gèrent l'état et valident si l'action a été exécutée ou non. Cela permet d'éviter qu'une étape soit exécutée plusieurs fois alors qu'elle est destinée à être exécutée une seule fois.

Voici un exemple de script idempotent qui redémarre l'instance plusieurs fois.

```
$name = Get current computer name
If ($name -ne $desiredName)
 {
  Rename computer
  exit 3010
 }

$domain = Get current domain name
```

```
If ($domain -ne $desiredDomain)
 {
  Join domain
  exit 3010
 }

If (desired package not installed)
 {
  Install package
  exit 3010
 }
```

Les échantillons de script suivants utilisent des codes de sortie pour redémarrer les instances. L'exemple sur Linux installe les mises à jour de package sur Amazon Linux, puis redémarre l'instance. L'exemple Windows installe le client Telnet sur l'instance, puis redémarre l'instance.

Amazon Linux exemple

```
#!/bin/bash
yum -y update
needs-restarting -r
if [ $? -eq 1 ]
then
        exit 194
else
        exit 0
fi
```

Exemple Windows

```
$telnet = Get-WindowsFeature -Name Telnet-Client
if (-not $telnet.Installed)
 {
  # Install Telnet and then send a reboot request to SSM Agent.
  Install-WindowsFeature -Name "Telnet-Client"
  exit 3010
 }
```

Gestion des codes de sortie dans les commandes Fonctionnalité Exécuter la commande

La fonctionnalité Fonctionnalité Exécuter la commande vous permet de définir la façon dont les codes de sortie sont gérés dans vos scripts. Par défaut, le code de sortie de la dernière commande exécutée dans un script est signalé comme le code de sortie pour l'ensemble du script. Par exemple, prenons un script qui contient trois commandes. La première échoue, mais les suivantes sont réalisées avec succès. Compte tenu du succès de la commande finale, l'état de l'exécution est signalé comme `succeeded`.

Scripts shell

Pour faire échouer la totalité de la commande lors du premier échec, vous pouvez inclure une instruction conditionnelle shell pour quitter le script si une commande précédant la dernière échoue. Utilisez pas l'approche suivante.

```
<command 1>
if [ $? != 0 ]
then
    exit <N>
fi
<command 2>
<command 3>
```

Dans l'exemple suivant, la totalité du script échoue si la première commande échoue.

```
cd /test
if [ $? != 0 ]
then
    echo "Failed"
    exit 1
fi
date
```

PowerShellScripts

PowerShellexige que vous appeliez `exit` explicitement dans vos scripts pour Fonctionnalité Exécuter la commande afin de capturer correctement le code de sortie.

```
<command 1>
if ($?) {<do something>}
else {exit <N>}
<command 2>
<command 3>
exit <N>
```

Voici un exemple :

```
cd C:\
if ($?) {echo "Success"}
else {exit 1}
date
```

Comprendre les états des commandes

La fonctionnalité Systems Manager de Fonctionnalité Exécuter la commande génère des rapports avec des informations de statut détaillées sur les différents états qu'une commande connaît pendant le traitement et pour chaque instance qui a traité la commande. Vous pouvez surveiller les statuts de commande à l'aide des méthodes suivantes.

- Cliquez sur l'icône Actualiser sur la page Exécuter la commande de la console Amazon EC2.
- Appelez list-commands ou list-command-invocations à l'aide de l'AWS CLI. Ou appelez Get-SSMCommand ou Get-SSMCommandInvocation à l'aide des Outils AWS pour Windows PowerShell.
- Configurez EventBridge pour répondre aux changements d'état ou d'état.
- Configurez Amazon SNS pour envoyer des notifications pour toutes les modifications de statut ou pour des statuts spécifiques tels que Failed ou TimedOut .

Statut Fonctionnalité Exécuter la commande

La fonctionnalité Fonctionnalité Exécuter la commande génère des rapports avec des détails de statut pour trois domaines : plug-ins, appels et statut de commande général. Un plug-in est un bloc d'exécution de code qui est défini dans le document Systems Manager (SSM) de votre commande. Pour de plus amples informations sur les plug-ins, veuillez consulter Référence du plug-in de document Systems Manager (p. 1232).

Lorsque vous envoyez une commande à plusieurs instances en même temps, chaque copie de la commande qui cible chaque instance correspond à un appel de commande. Par exemple, si vous utilisez le document AWS-RunShellScript et que vous envoyez une commande `ifconfig` à 20 instances, cette commande aura 20 appels. Chaque appel de commande signale le statut individuellement. Les plug-ins d'un appel de commande donné communiquent aussi le statut individuellement.

Enfin, la fonctionnalité Fonctionnalité Exécuter la commande inclut un statut de commande regroupé pour tous les plug-ins et les appels. Le statut de commande regroupé peut être différent du statut signalé par plug-ins ou appels, comme indiqué dans les tableaux suivants.

> **Note**
>
> Si vous exécutez des commandes sur un grand nombre d'instances à l'aide des paramètres `max-concurrency` ou `max-errors`, le statut de commande reflète les limites imposées par ces paramètres, comme décrit dans les tableaux suivants. Pour obtenir plus d'informations sur ces paramètres, consultez Utilisation des contrôles de cibles et de taux pour envoyer des commandes à un parc (p. 955).

Statut détaillé pour des plug-ins et des appels de commande

Statut	Details (Détails)
En suspens	La commande n'a pas encore été reçue par l'agent sur l'instance. Si la commande n'est pas reçue par l'agent avant que la valeur spécifiée par le paramètre Timeout (seconds) [Délai (secondes)] ne soit atteinte, le statut devient `Delivery Timed Out`.
InProgress	La commande a été reçue par l'agent ou son exécution a commencé sur l'instance. En fonction du résultat de tous les plug-ins de commande, le statut devient `Success` , ou `Failed` .`Execution Timed Out` Si l'agent n'est pas disponible sur l'instance, le statut de la commande s'affiche `InProgress` jusqu'à ce que l'agent soit à nouveau disponible. Le statut est ensuite remplacé par un état de mise hors service.
Delayed (Retardé)	Le système tente d'envoyer la commande à l'instance, mais n'a pas réussi. Le système va réessayer.
Success (Succès)	La commande a été reçue par Agent SSM sur l'instance et a retourné un code de sortie de zéro. Ce statut ne signifie pas que la commande a été effectuée avec succès sur l'instance. Il s'agit d'un état final. **Note** Pour résoudre les erreurs ou obtenir plus d'informations sur l'exécution des commandes, envoyez une commande qui gère les erreurs ou les exceptions en retournant les codes de sortie appropriés (codes de sortie autre que zéro pour un échec de la commande).
Delivery Timed Out (Délai dépassé)	La commande n'a pas été fournie à l'instance avant l'expiration du délai de remise. Les délais de remise ne sont pas pris en compte dans la `max-errors` limite de la commande parent, mais ils permettent de savoir si le statut de la commande parent est `Success` ou `Incomplete` . Il s'agit d'un état final.

Statut	Details (Détails)
Execution Timed Out (Expiration du délai d'exécution)	L'exécution de la commande a démarré sur l'instance, mais l'exécution ne s'est pas terminée avant l'expiration du délai d'exécution. Les dépassements de délai d'exécution sont comptabilisés dans la limite `max-errors` de la commande parent. Il s'agit d'un état final. Une fois le délai atteint, Systems Manager arrête l'exécution de la commande.
Failed (Échec)	La commande n'a pas réussi sur l'instance. Pour un plug-in, cela signifie que le code de résultat n'était pas zéro. Pour un appel de commande, cela signifie que le code de résultat pour un ou plusieurs plug-ins n'était pas zéro. Les échecs d'appel sont comptabilisés dans la limite `max-errors` de la commande parent. Il s'agit d'un état final.
Canceled (Annulé)	La commande a pris fin avant de se terminer. Il s'agit d'un état final.
Undeliverable (Non livrable)	La commande ne peut pas être délivrée à l'instance. L'instance peut ne pas exister ou ne peut pas répondre. Les appels non livrables ne sont pas pris en compte dans la `max-errors` limite de la commande parent, mais ils permettent de savoir si le statut de la commande parent est `Success` ou `Incomplete` . Par exemple, si tous les appels d'une commande ont le statut `Undeliverable` , le statut de la commande renvoyé est `Failed` . Toutefois, si une commande comporte cinq appels, dont quatre renvoient le statut `Undeliverable` et un autre renvoie le statut `Success` , le statut de la commande parent est `Success` . Il s'agit d'un état final.
Terminated (Résilié)	La commande parent atteint sa limite `max-errors` et les appels de commande suivants ont été annulés par le système. Il s'agit d'un état final.
Invalid Platform (Plateforme non valide)	La commande a été envoyée à une instance qui ne correspondait pas aux plateformes requises spécifiées par le document choisie. InvalidPlatformn'est pas pris en compte dans la limite max-errors de la commande parent, mais contribue à déterminer si le statut de la commande parent est Success ou Failed. Par exemple, si tous les appels d'une commande ont le statut `Invalid Platform` , le statut de la commande renvoyé est `Failed` . Toutefois, si une commande comporte cinq appels, dont quatre renvoient le statut `Invalid Platform` et un autre renvoie le statut `Success` , le statut de la commande parent est `Success` . Il s'agit d'un état final.

Statut	Details (Détails)
Access Denied (Accès refusé)	L'utilisateur ou le rôle IAM qui initie la commande n'a pas accès à l'instance gérée ciblée. `AccessDenied`n'est pas pris en compte dans la `max-errors` limite de la commande parent, mais contribue à déterminer si le statut de la commande parent est `Success` ou `Failed` . Par exemple, si tous les appels d'une commande ont le statut `AccessDenied` , le statut de la commande renvoyé est `Failed` . Toutefois, si une commande comporte cinq appels, dont quatre renvoient le statut `AccessDenied` et un autre renvoie le statut `Success` , le statut de la commande parent est `Success` . Il s'agit d'un état final.

Statut détaillé d'une commande

Statut	Details (Détails)
En suspens	La commande n'a pas encore été reçue par un agent sur une instance.
InProgress	La commande a été envoyée au moins à une instance, mais n'a pas atteint un état final sur toutes les instances.
Delayed (Retardé)	Le système tente d'envoyer la commande à l'instance, mais n'a pas réussi. Le système va réessayer.
Success (Succès)	La commande a été reçue par Agent SSM sur l'ensemble des instances spécifiées ou ciblées et a retourné un code de sortie de zéro. L'ensemble des invocations de commande a atteint un état de mise hors service, et la valeur de `max-errors` n'a pas été atteinte. Ce statut ne signifie pas que la commande a été effectuée avec succès sur l'ensemble des instances spécifiées ou ciblées. Il s'agit d'un état final. Note Pour résoudre les erreurs ou obtenir plus d'informations sur l'exécution des commandes, envoyez une commande qui gère les erreurs ou les exceptions en retournant les codes de sortie appropriés (codes de sortie autre que zéro pour un échec de la commande).
Delivery Timed Out (Délai dépassé)	La commande n'a pas été fournie à l'instance avant l'expiration du délai de remise. La valeur d'`max-errors`un ou plusieurs appels de commande indique le statut `Delivery Timed Out` . Il s'agit d'un état final.

Statut	Details (Détails)
Execution Timed Out (Expiration du délai d'exécution)	L'exécution de la commande a démarré sur l'instance, mais l'exécution ne s'est pas terminée avant l'expiration du délai d'exécution. La valeur d'`max-errors`un ou plusieurs appels de commande indique le statut `Execution Timed Out` . Il s'agit d'un état final.
Failed (Échec)	La commande n'a pas réussi sur l'instance. La valeur d'`max-errors`un ou plusieurs appels de commande indique le statut `Failed` . Il s'agit d'un état final.
Incomplete (Incomplet)	La commande a été tentée sur toutes les instances et un ou plusieurs des appels n'ont pas la valeur `Success` . Cependant, le nombre d'appels en échec n'est pas suffisant pour que le statut soit `Failed` . Il s'agit d'un état final.
Canceled (Annulé)	La commande a pris fin avant de se terminer. Il s'agit d'un état final.
Rate Exceeded (Taux dépassé)	Le nombre d'instances ciblées par la commande a dépassé la limite du compte pour les appels en attente. Le système a annulé la commande avant de l'exécuter sur une instance. Il s'agit d'un état final.
Access Denied (Accès refusé)	L'utilisateur ou le rôle IAM qui initie la commande n'a pas accès au groupe de ressources ciblé. `AccessDenied`n'est pas pris en compte dans la `max-errors` limite de la commande parent, mais contribue à déterminer si le statut de la commande parent est `Success` ou `Failed` . (Par exemple, si tous les appels d'une commande ont le statut `AccessDenied` , le statut de la commande renvoyé est `Failed` . Toutefois, si une commande comporte 5 appels, dont 4 renvoient le statut `AccessDenied` et 1 renvoie le statut `Success` , le statut de la commande parent est `Success` .) Il s'agit d'un état final.
No Instances In Tag (Aucune instance dans la balise)	La valeur ou le groupe de ressources de la paire de clés de balise ciblés par la commande ne correspondent à aucune instance gérée. Il s'agit d'un état final.

Présentation des valeurs de délai des commandes

Systems Manager applique les valeurs de délai suivantes lors de l'exécution des commandes.

Delivery Timeout (Délai de remise)

Dans la console Systems Manager, vous spécifiez la valeur du délai de remise dans le champ Timeout (seconds) [Délai (secondes)]. Systems Manager doit transmettre la commande aux cibles et l'Agent SSM

sur les cibles doit commencer à traiter la commande avant la fin du délai en secondes spécifié dans le champ Timeout (seconds) [Délai (secondes)].

Execution Timeout (Délai d'exécution)

Dans la console Systems Manager, vous spécifiez la valeur du délai d'exécution dans le champ Execution Timeout (Délai d'exécution) s'il est disponible. Les documents SSM ne nécessitent pas tous que vous spécifiiez un délai d'exécution. Si un délai est spécifié, la commande doit être exécutée dans ce délai.

Default Execution Timeout (Délai d'exécution par défaut)

Si un document Systems Manager (SSM) ne nécessite pas que vous spécifiiez explicitement une valeur de délai d'exécution, Systems Manager applique le délai d'exécution par défaut codé en dur.

Total Timeout (Délai total)

Le délai total est égal à la valeur de `delivery timeout` plus `execution timeout` . Si `nexecution timeout`'est pas requis par le SSM document, `total timeout` est égal à la valeur de `delivery timeout` plus `default execution timeout` .

Mode de signalisation des délais par Systems Manager

Si Systems Manager reçoit une réponse `execution timeout` de l'Agent SSM sur une cible, Systems Manager marque l'appel de commande comme `executionTimeout`.

Si Systems Manager ne reçoit aucune réponse de l'Agent SSM sur une cible et que l'exécution de la commande dépasse la valeur `total timeout`, Systems Manager marque l'appel de commande comme `deliveryTimeout`.

Afin de déterminer le statut du délai d'expiration sur une cible, Agent SSM combine tous les paramètres et le contenu du SSM document pour calculer pour `executionTimeout` . Lorsqu' Agent SSM détermine qu'une commande a expiré, il envoie `executionTimeout` au service .

> **Note**
>
> L'Agent SSM traite `executionTimeout` différemment selon le type de document SSM et la version du document.

Procédures Fonctionnalité Exécuter la commande

Les procédures de cette section vous montrent comment exécuter des commandes avec la fonctionnalité Fonctionnalité Exécuter la commande à l'aide de l'AWS Command Line Interface ou des Outils AWS pour Windows PowerShell.

Table des matières

- Procédure : Utilisation de l' AWS CLI avecFonctionnalité Exécuter la commande (p. 971)
- Procédure : Utilisation de l' Outils AWS pour Windows PowerShell avecFonctionnalité Exécuter la commande (p. 977)

Vous pouvez également consulter des exemples de commandes dans les références suivantes.

- Systems Manager de référence de l'interface de ligne de commande AWS
- Systems Manager Outils AWS pour Windows PowerShell Référence

Procédure : Utilisation de l' AWS CLI avecFonctionnalité Exécuter la commande

L'exemple de procédure suivant vous explique comment utiliser l'AWS CLI pour afficher les informations sur les commandes et leurs paramètres, comment exécuter les commandes et comment afficher le statut de ces dernières.

> **Important**
>
> Seuls les administrateurs de confiance doivent être autorisés à utiliser les documents préconfigurés Systems Manager illustrés dans cette rubrique. Les commandes ou scripts spécifiés dans des documents Systems Manager sont exécutés avec un privilège administratif sur vos instances. Si un utilisateur a l'autorisation d'exécuter l'un des documents Systems Manager prédéfinis (tout document qui commence par `AWS-`), cet utilisateur dispose d'un accès administrateur à l'instance. Pour tous les autres utilisateurs, vous devez créer des documents restrictifs et les partager avec des utilisateurs spécifiques. Pour de plus amples informations sur la restriction de l'accès à la fonctionnalité Fonctionnalité Exécuter la commande, veuillez consulter Créer des utilisateurs et des groupes IAM non administrateurs pour Systems Manager (p. 30).

Rubriques

- Étape 1 : Mise en route (p. 971)
- Étape 2 : Exécuter des scripts shell pour afficher les détails des ressources (p. 972)
- Étape 3 : Envoi de commandes simples à l'aide du document AWS-RunShellScript (p. 973)
- Étape 4 : Exécuter un script Python simple à l'aide de l'Fonctionnalité Exécuter la commande (p. 975)
- Étape 5 : Exécution d'un script Bash à l'aide d'Fonctionnalité Exécuter la commande (p. 975)

Étape 1 : Mise en route

Vous devez disposer de privilèges administrateur sur les instances que vous souhaitez configurer ou vous devez bénéficier de l'autorisation appropriée dans IAM. Notez également que cet exemple utilise la région Région USA Est (Ohio) (us-east-2). Fonctionnalité Exécuter la commande est actuellement disponible dans les régions AWS répertoriées dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services. Pour plus d'informations, consultez la section Prérequis d'Systems Manager (p. 15).

Pour exécuter des commandes à l'aide de l'AWS CLI

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).
2. Répertorier tous les documents disponibles

   Cette commande répertorie tous les documents disponibles pour votre compte en fonction des autorisations IAM.

```
aws ssm list-documents
```

3. Vérifier qu'une instance est prête à recevoir les commandes

   La sortie de la commande suivante indique si les instances sont en ligne.

   Linux

   ```
   aws ssm describe-instance-information \
    --output text --query "InstanceInformationList[*]"
   ```

   Windows

   ```
   aws ssm describe-instance-information ^
    --output text --query "InstanceInformationList[*]"
   ```

4. Utilisez la commande suivante pour afficher des détails sur une instance spécifique.

   > **Note**
   >
   > Pour exécuter les commandes de cette procédure, vous devez remplacer l'instance et la commande IDs . L'ID de commande est renvoyé en réponse à send-command . L'ID d'instance est disponible à partir de la Amazon EC2 console .

   Linux

   ```
   aws ssm describe-instance-information \
    --instance-information-filter-list key=InstanceIds,valueSet=instance-ID
   ```

   Windows

   ```
   aws ssm describe-instance-information ^
    --instance-information-filter-list key=InstanceIds,valueSet=instance-ID
   ```

Étape 2 : Exécuter des scripts shell pour afficher les détails des ressources

La fonctionnalité Fonctionnalité Exécuter la commande et le document AWS-RunShellScript vous permettent d'exécuter n'importe quel script ou commande sur une instance EC2 comme si vous vous étiez connecté localement.

Afficher la description et les paramètres disponibles

Utilisez la commande suivante pour afficher la description du document JSON Systems Manager.

Linux

```
aws ssm describe-document \
 --name "AWS-RunShellScript" \
 --query "[Document.Name,Document.Description]"
```

Windows

```
aws ssm describe-document ^
 --name "AWS-RunShellScript" ^
 --query "[Document.Name,Document.Description]"
```

Utilisez la commande suivante afin d'afficher les paramètres disponibles et les détails les concernant.

Linux

```
aws ssm describe-document \
 --name "AWS-RunShellScript" \
 --query "Document.Parameters[*]"
```

Windows

```
aws ssm describe-document ^
 --name "AWS-RunShellScript" ^
 --query "Document.Parameters[*]"
```

Étape 3 : Envoi de commandes simples à l'aide du document AWS-RunShellScript

Utilisez la commande suivante afin d'obtenir des informations IP pour une instance.

Linux

```
aws ssm send-command \
 --instance-ids "instance-ID" \
 --document-name "AWS-RunShellScript" \
 --comment "IP config" \
 --parameters commands=ifconfig \
 --output text
```

Windows

```
aws ssm send-command ^
 --instance-ids "instance-ID" ^
 --document-name "AWS-RunShellScript" ^
 --comment "IP config" ^
 --parameters commands=ifconfig ^
 --output text
```

Obtenir des informations sur la commande avec des données de réponse

La commande suivante utilise l'ID de commande qui a été retourné par la commande précédente afin d'obtenir les détails et les données de réponse de l'exécution de la commande. Le système renvoie les données de réponse si la commande a été exécutée. Si l'exécution de la commande indique `"Pending"` ou `"InProgress"`, vous devrez l'exécuter à nouveau pour consulter les données de réponse.

Linux

```
aws ssm list-command-invocations \
 --command-id $sh-command-id \
 --details
```

Windows

```
aws ssm list-command-invocations ^
 --command-id $sh-command-id ^
 --details
```

Identifier le compte utilisateur

La commande suivante affiche le compte utilisateur par défaut qui exécute les commandes.

Linux

```
sh_command_id=$(aws ssm send-command \
 --instance-ids "instance-ID" \
 --document-name "AWS-RunShellScript" \
 --comment "Demo run shell script on Linux Instance" \
 --parameters commands=whoami \
 --output text \
 --query "Command.CommandId")
```

Obtenir le statut de la commande

La commande suivante utilise l'ID de commande afin d'obtenir le statut de l'exécution de la commande sur l'instance. Cet exemple utilise l'ID de commande qui a été renvoyé lors de la commande précédente.

Linux

```
aws ssm list-commands \
 --command-id "command-ID"
```

Windows

```
aws ssm list-commands ^
 --command-id "command-ID"
```

Obtenir les détails de la commande

La commande suivante utilise l'ID de la commande précédente afin d'obtenir le statut de l'exécution de la commande par instance.

Linux

```
aws ssm list-command-invocations \
 --command-id "command-ID" \
 --details
```

Windows

```
aws ssm list-command-invocations ^
 --command-id "command-ID" ^
 --details
```

Obtenir des informations sur la commande avec des données de réponse pour une instance spécifique

La commande suivante renvoie la sortie de la demande `aws ssm send-command` initiale pour une instance spécifique.

Linux

```
aws ssm list-command-invocations \
 --instance-id instance-ID \
 --command-id "command-ID" \
 --details
```

Windows

```
aws ssm list-command-invocations ^
 --instance-id instance-ID ^
 --command-id "command-ID" ^
 --details
```

Afficher la version Python

La commande suivante retourne la version de Python exécutée sur une instance.

Linux

```
sh_command_id=$(aws ssm send-command \
 --instance-ids "instance-ID" \
 --document-name "AWS-RunShellScript" \
 --comment "Demo run shell script on Linux Instances" \
 --parameters commands='python -V' \
 --output text --query "Command.CommandId") sh -c 'aws ssm list-command-invocations \
 --command-id "$sh_command_id" \
 --details \
 --query "CommandInvocations[].CommandPlugins[].{Status:Status,Output:Output}"'
```

Étape 4 : Exécuter un script Python simple à l'aide de l'Fonctionnalité Exécuter la commande

La commande suivante exécute un simple script Python « Hello World » à l'aide de Fonctionnalité Exécuter la commande.

Linux

```
sh_command_id=$(aws ssm send-command \
 --instance-ids "instance-ID" \
 --document-name "AWS-RunShellScript" \
 --comment "Demo run shell script on Linux Instances" \
 --parameters '{"commands":["#!/usr/bin/python","print \"Hello world from python\""]}'
 \
 --output text \
 --query "Command.CommandId") sh -c 'aws ssm list-command-invocations \
 --command-id "$sh_command_id" \
 --details \
 --query "CommandInvocations[].CommandPlugins[].{Status:Status,Output:Output}"'
```

Étape 5 : Exécution d'un script Bash à l'aide d'Fonctionnalité Exécuter la commande

Les exemples de cette section montrent comment exécuter le script bash suivant à l'aide d' Fonctionnalité Exécuter la commande .

Pour obtenir des exemples d'utilisation d' Fonctionnalité Exécuter la commande pour exécuter des scripts stockés dans des emplacements distants, consultez Exécution de scripts depuis Amazon S3 (p. 126) et Exécution de scripts depuis GitHub (p. 139) .

```
#!/bin/bash
yum -y update
yum install -y ruby
```

```
cd /home/ec2-user
curl -O https://aws-codedeploy-us-east-2.s3.amazonaws.com/latest/install
chmod +x ./install
./install auto
```

Ce script installe lAWS CodeDeploy'agent sur les instances Amazon Linux et Red Hat Enterprise Linux (RHEL), comme décrit dans Créer une Amazon EC2 instance pour CodeDeploy dans le AWS CodeDeploy User Guide .

Le script installe lCodeDeploy'agent à partir d'un Amazon S3 compartiment géré par AWS dans l' Région USA Est (Ohio) (us-east-2) `aws-codedeploy-us-east-2` .

Exécuter un script bash dans une AWS CLI commande de l'

L'exemple suivant montre comment inclure le script bash dans une commande d'interface de ligne de commande à l'aide de l'`--parameters`option .

Linux

```
aws ssm send-command \
 --document-name "AWS-RunShellScript" \
 --targets '[{"Key":"InstanceIds","Values":["instance-id"]}]' \
 --parameters '{"commands":["#!/bin/bash","yum -y update","yum install -y ruby","cd /
home/ec2-user","curl -O https://aws-codedeploy-us-east-2.s3.amazonaws.com/latest/
install","chmod +x ./install","./install auto"]}'
```

Exécuter un script bash dans un fichier JSON

Dans l'exemple suivant, le contenu du script bash est stocké dans un fichier JSON et le fichier est inclus dans la commande à l'aide de l'`--cli-input-json`option .

La commande :

Linux

```
aws ssm send-command \
 --document-name "AWS-RunShellScript" \
 --targets "Key=InstanceIds,Values=instance-id" \
 --cli-input-json file://installCodeDeployAgent.json
```

Windows

```
aws ssm send-command ^
 --document-name "AWS-RunShellScript" ^
 --targets "Key=InstanceIds,Values=instance-id" ^
 --cli-input-json file://installCodeDeployAgent.json
```

Le contenu du `installCodeDeployAgent.json` fichier référencé :

```
{
    "Parameters": {
        "commands": [
            "#!/bin/bash",
            "yum -y update",
            "yum install -y ruby",
            "cd /home/ec2-user",
            "curl -O https://aws-codedeploy-us-east-2.s3.amazonaws.com/latest/install",
            "chmod +x ./install",
```

```
            "./install auto"
        ]
    }
}
```

Procédure : Utilisation de l' Outils AWS pour Windows PowerShell avecFonctionnalité Exécuter la commande

Les exemples suivants vous expliquent comment utiliser les Outils pour Windows PowerShell pour afficher les informations sur les commandes et leurs paramètres, comment exécuter les commandes et comment afficher le statut de ces dernières. Cette procédure inclut un exemple pour chaque document Systems Manager prédéfini.

> **Important**
>
> Seuls les administrateurs de confiance doivent être autorisés à utiliser les documents préconfigurés Systems Manager illustrés dans cette rubrique. Les commandes ou scripts spécifiés dans des documents Systems Manager sont exécutés avec un privilège administratif sur vos instances. Si un utilisateur a l'autorisation d'exécuter l'un des documents Systems Manager prédéfinis (tout document qui commence par AWS), cet utilisateur dispose d'un accès administrateur à l'instance. Pour tous les autres utilisateurs, vous devez créer des documents restrictifs et les partager avec des utilisateurs spécifiques. Pour de plus amples informations sur la restriction de l'accès à la fonctionnalité Fonctionnalité Exécuter la commande, veuillez consulter Créer des utilisateurs et des groupes IAM non administrateurs pour Systems Manager (p. 30).

Rubriques

- Configuration des paramètres de session Outils AWS pour Windows PowerShell (p. 977)
- Répertorier tous les documents disponibles (p. 978)
- Exécuter PowerShell des commandes ou des scripts (p. 978)
- Installer une application à l'aide du document AWS-InstallApplication (p. 979)
- Installer un PowerShell module à l'aide du document JSON AWS-InstallPowerShellModule (p. 980)
- Joindre une instance à un domaine à l'aide du document JSON AWS-JoinDirectoryServiceDomain (p. 981)
- Envoi de métriques Windows à à Amazon CloudWatch Logs l'aide du document AWS-ConfigureCloudWatch (p. 982)
- Mettre à jour EC2Config à l'aide du document AWS-UpdateEC2Config (p. 983)
- Activer/Désactiver les mises à jour automatiques Windows à l'aide du document AWS-ConfigureWindowsUpdate (p. 984)
- Gérer les mises à jour Windows à l'aide de la fonctionnalité Fonctionnalité Exécuter la commande (p. 985)

Configuration des paramètres de session Outils AWS pour Windows PowerShell

Spécifier vos informations d'identification

Ouvrez les Outils AWS pour Windows PowerShell sur votre ordinateur local et exécutez la commande suivante pour spécifier vos informations d'identification. Vous devez disposer de privilèges administrateur sur les instances que vous souhaitez configurer ou vous devez bénéficier de l'autorisation appropriée dans IAM. Pour plus d'informations, consultez la section Prérequis d'Systems Manager (p. 15).

```
Set-AWSCredentials -AccessKey key-name -SecretKey key-name
```

Définir une région AWS par défaut

Exécutez la commande suivante pour définir la région pour votre PowerShell session. Notez également que cet exemple utilise la région Région USA Est (Ohio) (us-east-2). Fonctionnalité Exécuter la commande est actuellement disponible dans les régions AWS répertoriées dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

```
Set-DefaultAWSRegion `
 -Region us-east-2
```

Répertorier tous les documents disponibles

Cette commande répertorie tous les documents disponibles pour votre compte :

```
Get-SSMDocumentList
```

Exécuter PowerShell des commandes ou des scripts

La fonctionnalité Fonctionnalité Exécuter la commande et le document `AWS-RunPowerShell` vous permettent d'exécuter n'importe quel script ou commande sur une instance EC2 comme si vous étiez connecté à cette instance via le Bureau à distance. Vous pouvez émettre des commandes ou entrer un chemin d'accès à un script local pour exécuter la commande.

> **Note**
>
> Pour de plus amples informations sur le redémarrage des serveurs et des instances en appelant des scripts à l'aide de Fonctionnalité Exécuter la commande, veuillez consulter Redémarrage d'une instance gérée à partir de scripts (p. 963).

Afficher la description et les paramètres disponibles

```
Get-SSMDocumentDescription `
 -Name "AWS-RunPowerShellScript"
```

Afficher des informations supplémentaires sur les paramètres

```
Get-SSMDocumentDescription `
 -Name "AWS-RunPowerShellScript" | Select -ExpandProperty Parameters
```

Envoyer une commande à l'aide du document AWS-RunPowerShellScript

La commande suivante permet d'afficher le contenu du répertoire `"C:\Users"` et celui du répertoire `"C:\"` sur deux instances.

```
$runPSCommand = Send-SSMCommand `
 -InstanceIds @("instance-ID-1", "instance-ID-2") `
 -DocumentName "AWS-RunPowerShellScript" `
 -Comment "Demo AWS-RunPowerShellScript with two instances" `
 -Parameter @{'commands'=@('dir C:\Users', 'dir C:\')}
```

Obtenir les détails de la demande de commande

La commande suivante utilise l'`CommandId` afin d'obtenir le statut de l'exécution de la commande sur les deux instances. Cet exemple utilise l'`CommandId` qui a été renvoyé lors de la commande précédente.

```
Get-SSMCommand `
 -CommandId $runPSCommand.CommandId
```

Le statut de la commande figurant dans cet exemple peut être Success, Pending ou InProgress.

Obtenir des informations sur la commande par instance

La commande suivante utilise l'`CommandId` de la commande précédente afin d'obtenir le statut de l'exécution de la commande par instance.

```
Get-SSMCommandInvocation `
 -CommandId $runPSCommand.CommandId
```

Obtenir des informations sur la commande avec des données de réponse pour une instance spécifique

La commande suivante renvoie la sortie de la commande `Send-SSMCommand` initiale pour une instance spécifique.

```
Get-SSMCommandInvocation `
 -CommandId $runPSCommand.CommandId `
 -Details $true `
 -InstanceId instance-ID | Select -ExpandProperty CommandPlugins
```

Annuler une commande

La commande suivante annule le `Send-SSMCommand` pour le document `AWS-RunPowerShellScript`.

```
$cancelCommand = Send-SSMCommand `
 -InstanceIds @("instance-ID-1","instance-ID-2") `
 -DocumentName "AWS-RunPowerShellScript" `
 -Comment "Demo AWS-RunPowerShellScript with two instances" `
 -Parameter @{'commands'='Start-Sleep -Seconds 120; dir C:\'}

Stop-SSMCommand -CommandId $cancelCommand.CommandId
```

Vérifier le statut de commande

La commande suivante vérifie le statut de la commande `Cancel`.

```
Get-SSMCommand `
 -CommandId $cancelCommand.CommandId
```

Installer une application à l'aide du document AWS-InstallApplication

En utilisant Fonctionnalité Exécuter la commande et le document `AWS-InstallApplication`, vous pouvez installer, réparer ou désinstaller des applications sur des instances. Cette commande a besoin d'un chemin d'accès ou d'une adresse pour un MSI.

> **Note**
>
> Pour de plus amples informations sur le redémarrage des serveurs et des instances en appelant des scripts à l'aide de Fonctionnalité Exécuter la commande, veuillez consulter Redémarrage d'une instance gérée à partir de scripts (p. 963).

Afficher la description et les paramètres disponibles

```
Get-SSMDocumentDescription `
 -Name "AWS-InstallApplication"
```

Afficher des informations supplémentaires sur les paramètres

```
Get-SSMDocumentDescription `
 -Name "AWS-InstallApplication" | Select -ExpandProperty Parameters
```

Envoyer une commande à l'aide du document AWS-InstallApplication

La commande suivante installe une version de Python sur votre instance en mode sans surveillance et consigne la sortie dans un fichier texte local sur votre lecteur `C:`.

```
$installAppCommand = Send-SSMCommand `
 -InstanceId instance-ID `
 -DocumentName "AWS-InstallApplication" `
 -Parameter @{'source'='https://www.python.org/ftp/python/2.7.9/python-2.7.9.msi';
 'parameters'='/norestart /quiet /log c:\pythoninstall.txt'}
```

Obtenir des informations sur la commande par instance

La commande suivante utilise l'`CommandId` afin d'obtenir le statut de l'exécution de la commande

```
Get-SSMCommandInvocation `
 -CommandId $installAppCommand.CommandId `
 -Details $true
```

Obtenir des informations sur la commande avec des données de réponse pour une instance spécifique

La commande suivante retourne les résultats de l'installation Python.

```
Get-SSMCommandInvocation `
 -CommandId $installAppCommand.CommandId `
 -Details $true `
 -InstanceId instance-ID | Select -ExpandProperty CommandPlugins
```

Installer un PowerShell module à l'aide du document JSON AWS-InstallPowerShellModule

Vous pouvez utiliser Fonctionnalité Exécuter la commande pour installer PowerShell des modules sur une instance EC2. Pour plus d'informations sur PowerShell les modules , consultez Modules PowerShell Windows.

Afficher la description et les paramètres disponibles

```
Get-SSMDocumentDescription `
 -Name "AWS-InstallPowerShellModule"
```

Afficher des informations supplémentaires sur les paramètres

```
Get-SSMDocumentDescription `
 -Name "AWS-InstallPowerShellModule" | Select -ExpandProperty Parameters
```

Installer un module PowerShell

La commande suivante télécharge le fichier EZOut .zip, l'installe, puis exécute une commande supplémentaire pour installer XPS Viewer. Enfin, la sortie de cette commande est chargée dans un compartiment S3 nommé demo-ssm-output-bucket.

```
$installPSCommand = Send-SSMCommand `
 -InstanceId instance-ID `
 -DocumentName "AWS-InstallPowerShellModule" `
 -Parameter @{'source'='https://gallery.technet.microsoft.com/EZOut-33ae0fb7/file/110351/1/
EZOut.zip';'commands'=@('Add-WindowsFeature -name XPS-Viewer -restart')} `
 -OutputS3BucketName demo-ssm-output-bucket
```

Obtenir des informations sur la commande par instance

La commande suivante utilise l'`CommandId` afin d'obtenir le statut de l'exécution de la commande

```
Get-SSMCommandInvocation `
 -CommandId $installPSCommand.CommandId `
 -Details $true
```

Obtenir des informations sur la commande avec des données de réponse pour l'instance

La commande suivante renvoie la sortie de la commande `Send-SSMCommand` d'origine pour le `CommandId` spécifique.

```
Get-SSMCommandInvocation `
 -CommandId $installPSCommand.CommandId `
 -Details $true | Select -ExpandProperty CommandPlugins
```

Joindre une instance à un domaine à l'aide du document JSON AWS-JoinDirectoryServiceDomain

La fonctionnalité Fonctionnalité Exécuter la commande vous permet de joindre rapidement une instance à un domaine AWS Directory Service. Avant d'exécuter cette commande, vous devez créer un répertoire. Nous vous recommandons également de vous familiariser avec l'AWS Directory Service. Pour plus d'informations, consultez Qu'est-ce que AWS Directory Service ?

À l'heure actuelle, vous pouvez uniquement associer une instance à un domaine. Vous ne pouvez pas supprimer d'instance d'un domaine.

> **Note**
>
> Pour de plus amples informations sur le redémarrage des serveurs et des instances en appelant des scripts à l'aide de Fonctionnalité Exécuter la commande, veuillez consulter Redémarrage d'une instance gérée à partir de scripts (p. 963).

Afficher la description et les paramètres disponibles

```
Get-SSMDocumentDescription `
 -Name "AWS-JoinDirectoryServiceDomain"
```

Afficher des informations supplémentaires sur les paramètres

```
Get-SSMDocumentDescription `
 -Name "AWS-JoinDirectoryServiceDomain" | Select -ExpandProperty Parameters
```

Joindre une instance à un domaine

La commande suivante joint l'instance au AWS Directory Service domaine donné et charge toute sortie générée dans le bucketS3 compartiment S3 de l'exemple.

```
$domainJoinCommand = Send-SSMCommand `
 -InstanceId instance-ID `
 -DocumentName "AWS-JoinDirectoryServiceDomain" `
 -Parameter @{'directoryId'='d-example01'; 'directoryName'='ssm.example.com';
 'dnsIpAddresses'=@('192.168.10.195', '192.168.20.97')} `
 -OutputS3BucketName demo-ssm-output-bucket
```

Obtenir des informations sur la commande par instance

La commande suivante utilise l'`CommandId` afin d'obtenir le statut de l'exécution de la commande

```
Get-SSMCommandInvocation `
 -CommandId $domainJoinCommand.CommandId `
 -Details $true
```

Obtenir des informations sur la commande avec des données de réponse pour l'instance

Cette commande renvoie la sortie du `Send-SSMCommand` initial pour le `CommandId` spécifique.

```
Get-SSMCommandInvocation `
 -CommandId $domainJoinCommand.CommandId `
 -Details $true | Select -ExpandProperty CommandPlugins
```

Envoi de métriques Windows à à Amazon CloudWatch Logs l'aide du document AWS-ConfigureCloudWatch

Vous pouvez envoyer Windows Server des messages dans les journaux de l'application, du système, de la sécurité et du suivi des événements pour Windows (ETW) vers Amazon CloudWatch Logs . Lorsque vous activez la journalisation pour la première fois, Systems Manager envoie tous les journaux générés en une minute à partir du moment où vous avez commencé à charger les journaux pour les applications, le système, la sécurité et le suivi d'événements. Les journaux générés avant ce moment ne sont pas inclus. Si vous désactivez la journalisation, puis que vous la réactivez, Systems Manager envoie les journaux à partir du moment où la journalisation a été désactivée. Pour tous les fichiers journaux personnalisés et les journaux IIS (Internet Information Services), Systems Manager lit les fichiers journaux depuis le début. En outre, Systems Manager peut également envoyer des données de compteur de performances à CloudWatch Logs .

Si vous avez déjà activé lCloudWatch'intégration dans EC2Config , les Systems Manager paramètres remplacent les paramètres stockés localement sur l'instance dans le fichier C:\\Program Files\\Amazon\\EC2ConfigService\\Settings\\AWS.EC2.Windows.CloudWatch.json. Pour plus d'informations sur l'utilisation d' EC2Config pour gérer les compteurs et journaux de performances sur une seule instance, consultez Envoi de compteurs CloudWatch de performances à et de journaux à LogsCloudWatch.

Afficher la description et les paramètres disponibles

```
Get-SSMDocumentDescription `
 -Name "AWS-ConfigureCloudWatch"
```

Afficher des informations supplémentaires sur les paramètres

```
Get-SSMDocumentDescription `
 -Name "AWS-ConfigureCloudWatch" | Select -ExpandProperty Parameters
```

Envoi de journaux d'application àCloudWatch

La commande suivante configure l'instance et transfère les journaux d'applications Windows dans CloudWatch.

```
$cloudWatchCommand = Send-SSMCommand `
 -InstanceID instance-ID `
 -DocumentName "AWS-ConfigureCloudWatch" `
 -Parameter @{'properties'='{"engineConfiguration":
 {"PollInterval":"00:00:15", "Components":[{"Id":"ApplicationEventLog",
 "FullName":"AWS.EC2.Windows.CloudWatch.EventLog.EventLogInputComponent,AWS.EC2.Windows.CloudWatch",
 "Parameters":{"LogName":"Application", "Levels":"7"}},{"Id":"CloudWatch",
 "FullName":"AWS.EC2.Windows.CloudWatch.CloudWatchLogsOutput,AWS.EC2.Windows.CloudWatch",
 "Parameters":{"Region":"region", "LogGroup":"my-log-group", "LogStream":"instance-id"}}],
 "Flows":{"Flows":["ApplicationEventLog,CloudWatch"]}}}'}
```

Obtenir des informations sur la commande par instance

La commande suivante utilise l'`CommandId` afin d'obtenir le statut de l'exécution de la commande

```
Get-SSMCommandInvocation `
 -CommandId $cloudWatchCommand.CommandId `
 -Details $true
```

Obtenir des informations sur la commande avec des données de réponse pour une instance spécifique

La commande suivante renvoie les résultats de la CloudWatch configuration Amazon.

```
Get-SSMCommandInvocation `
 -CommandId $cloudWatchCommand.CommandId `
 -Details $true `
 -InstanceId instance-ID | Select -ExpandProperty CommandPlugins
```

Envoi de compteurs de performances à à CloudWatch l'aide du document AWS-ConfigureCloudWatch

La commande de démonstration suivante charge les compteurs de performances dans CloudWatch. Pour de plus amples informations, veuillez consulter Guide de l'utilisateur Amazon CloudWatch.

```
$cloudWatchMetricsCommand = Send-SSMCommand `
 -InstanceID instance-ID `
 -DocumentName "AWS-ConfigureCloudWatch" `
 -Parameter @{'properties'='{"engineConfiguration":
 {"PollInterval":"00:00:15", "Components":[{"Id":"PerformanceCounter",
 "FullName":"AWS.EC2.Windows.CloudWatch.PerformanceCounterComponent.PerformanceCounterInputComponent,AW
 "Parameters":{"CategoryName":"Memory", "CounterName":"Available
 MBytes", "InstanceName":"", "MetricName":"AvailableMemory",
 "Unit":"Megabytes","DimensionName":"", "DimensionValue":""}},{"Id":"CloudWatch",
 "FullName":"AWS.EC2.Windows.CloudWatch.CloudWatch.CloudWatchOutputComponent,AWS.EC2.Windows.CloudWatch
 "Parameters":{"AccessKey":"", "SecretKey":"","Region":"region", "NameSpace":"Windows-
Default"}}], "Flows":{"Flows":["PerformanceCounter,CloudWatch"]}}}'}
```

Mettre à jour EC2Config à l'aide du document AWS-UpdateEC2Config

Avec Fonctionnalité Exécuter la commande et le `AWS-EC2ConfigUpdate` document , vous pouvez mettre à jour le EC2Config service en cours d'exécution sur vos Windows Server instances. Cette commande peut mettre à jour le EC2Config service avec la dernière version ou avec une version que vous spécifiez.

Afficher la description et les paramètres disponibles

```
Get-SSMDocumentDescription `
 -Name "AWS-UpdateEC2Config"
```

Afficher des informations supplémentaires sur les paramètres

```
Get-SSMDocumentDescription `
 -Name "AWS-UpdateEC2Config" | Select -ExpandProperty Parameters
```

Mise à jour EC2Config vers la dernière version

```
$ec2ConfigCommand = Send-SSMCommand `
 -InstanceId instance-ID `
 -DocumentName "AWS-UpdateEC2Config"
```

Obtenir des informations sur la commande avec des données de réponse pour l'instance

Cette commande renvoie la sortie de la commande spécifiée à partir de la commande `Send-SSMCommand` précédente :

```
Get-SSMCommandInvocation `
 -CommandId $ec2ConfigCommand.CommandId `
 -Details $true `
 -InstanceId instance-ID | Select -ExpandProperty CommandPlugins
```

Mise EC2Config à jour d'une version spécifique

La commande suivante permet de mettre à niveau EC2Config vers une version antérieure :

```
Send-SSMCommand `
 -InstanceId instance-ID `
 -DocumentName "AWS-UpdateEC2Config" `
 -Parameter @{'version'='4.9.3519'; 'allowDowngrade'='true'}
```

Activer/Désactiver les mises à jour automatiques Windows à l'aide du document AWS-ConfigureWindowsUpdate

La fonctionnalité Fonctionnalité Exécuter la commande et le document `AWS-ConfigureWindowsUpdate` vous permettent d'activer et de désactiver les mises à jour Windows automatiques sur vos instances Windows Server. Cette commande configure l'agent de mise à jour Windows pour télécharger et installer les mises à jour Windows à la date et à l'heure que vous spécifiez. Si une mise à jour requiert un redémarrage, l'ordinateur redémarre automatiquement 15 minutes après l'installation des mises à jour. Cette commande vous permet également de configurer la mise à jour Windows de façon à rechercher les mises à jour sans les installer. Le document `AWS-ConfigureWindowsUpdate` est compatible avec Windows Server 2008, 2008 R2, 2012, 2012 R2 et 2016.

Afficher la description et les paramètres disponibles

```
Get-SSMDocumentDescription `
 -Name "AWS-ConfigureWindowsUpdate"
```

Afficher des informations supplémentaires sur les paramètres

```
Get-SSMDocumentDescription `
 -Name "AWS-ConfigureWindowsUpdate" | Select -ExpandProperty Parameters
```

Activer la mise à jour automatique Windows

La commande suivante configure les mises à jour Windows de façon à ce qu'elles soient téléchargées et installées automatiquement tous les jours à 10 h.

```
$configureWindowsUpdateCommand = Send-SSMCommand `
 -InstanceId instance-ID `
 -DocumentName "AWS-ConfigureWindowsUpdate" `
 -Parameters @{'updateLevel'='InstallUpdatesAutomatically'; 'scheduledInstallDay'='Daily';
 'scheduledInstallTime'='22:00'}
```

Afficher le statut de la commande afin d'activer les mises à jour automatiques Windows

La commande suivante utilise l'`CommandId` afin d'obtenir le statut de l'exécution de la commande pour activer les mises à jour automatiques Windows.

```
Get-SSMCommandInvocation `
 -Details $true `
```

```
 -CommandId $configureWindowsUpdateCommand.CommandId | Select -ExpandProperty
 CommandPlugins
```

Désactiver la mise à jour automatique Windows

La commande suivante diminue le niveau de notification des mises à jour Windows afin que le système recherche les mises à jour, mais ne mette pas à jour l'instance automatiquement.

```
$configureWindowsUpdateCommand = Send-SSMCommand `
 -InstanceId instance-ID `
 -DocumentName "AWS-ConfigureWindowsUpdate" `
 -Parameters @{'updateLevel'='NeverCheckForUpdates'}
```

Afficher le statut de la commande afin de désactiver les mises à jour automatiques Windows

La commande suivante utilise l'`CommandId` afin d'obtenir le statut de l'exécution de la commande pour désactiver les mises à jour automatiques Windows.

```
Get-SSMCommandInvocation `
 -Details $true `
 -CommandId $configureWindowsUpdateCommand.CommandId | Select -ExpandProperty
 CommandPlugins
```

Gérer les mises à jour Windows à l'aide de la fonctionnalité Fonctionnalité Exécuter la commande

En utilisant Fonctionnalité Exécuter la commande et le document `AWS-InstallWindowsUpdates`, vous pouvez gérer les mises à jour pour les instances EC2 pour Windows Server. Cette commande recherche ou installe les mises à jour manquantes sur vos instances EC2 pour Windows Server et, éventuellement, provoque un redémarrage après l'installation. Vous pouvez également spécifier les classifications et les niveaux de gravité appropriés pour les mises à jour à installer dans votre environnement.

> **Note**
>
> Pour de plus amples informations sur le redémarrage des serveurs et des instances en appelant des scripts à l'aide de Fonctionnalité Exécuter la commande, veuillez consulter Redémarrage d'une instance gérée à partir de scripts (p. 963).

Les exemples suivants décrivent comment exécuter les tâches de gestion des mises à jour Windows spécifiées.

Rechercher toutes les mises à jour Windows manquantes

```
Send-SSMCommand `
 -InstanceId instance-ID `
 -DocumentName "AWS-InstallWindowsUpdates" `
 -Parameters @{'Action'='Scan'}
```

Installer des mises à jour Windows spécifiques

```
Send-SSMCommand `
 -InstanceId instance-ID `
 -DocumentName "AWS-InstallWindowsUpdates" `
 -Parameters @{'Action'='Install';'IncludeKbs'='kb-ID-1,kb-ID-2,kb-
ID-3';'AllowReboot'='True'}
```

Installer les mises à jour Windows importantes manquantes

```
Send-SSMCommand `
```

```
 -InstanceId instance-ID `
 -DocumentName "AWS-InstallWindowsUpdates" `
 -Parameters @{'Action'='Install';'SeverityLevels'='Important';'AllowReboot'='True'}
```

Installer les mises à jour Windows manquantes avec des exclusions spécifiques

```
Send-SSMCommand `
 -InstanceId instance-ID `
 -DocumentName "AWS-InstallWindowsUpdates" `
 -Parameters @{'Action'='Install';'ExcludeKbs'='kb-ID-1,kb-ID-2';'AllowReboot'='True'}
```

Dépannage de la fonctionnalité Exécuter la commande de Systems Manager

La fonctionnalité Fonctionnalité Exécuter la commande fournit les détails de l'état avec chaque exécution de commande. Pour de plus amples informations sur les détails de l'état des commande, veuillez consulter Comprendre les états des commandes (p. 965). Vous pouvez également utiliser les informations de cette rubrique pour aider à la résolution les problèmes rencontrés avec la fonctionnalité Fonctionnalité Exécuter la commande.

Rubriques

- Certaines de mes instances sont manquantes (p. 986)
- Une étape de mon script a échoué, mais l'état global est « réussi » (p. 986)
- Agent SSMne s'exécute pas correctement (p. 986)

Certaines de mes instances sont manquantes

Dans la page Exécuter une commande, une fois que vous avez choisi un document SSM à exécuter et que vous avez sélectionné Sélection manuelle des instances dans la section Cibles, la liste des instances sur lesquelles vous pouvez choisir d'exécuter la commande s'affiche.

Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.

> **Note**
>
> Après avoir créé, activé, redémarré ou relancé une instance gérée, installé Agent SSM sur une instance ou attaché un profil d'instance IAM à une instance, l'affichage de l'instance dans la liste peut prendre quelques minutes.

Une étape de mon script a échoué, mais l'état global est « réussi »

La fonctionnalité Fonctionnalité Exécuter la commande vous permet de définir la façon dont les codes de sortie sont gérés dans vos scripts. Par défaut, le code de sortie de la dernière commande exécutée dans un script est signalé comme le code de sortie pour l'ensemble du script. Vous pouvez néanmoins inclure une instruction conditionnelle pour quitter le script si une commande précédant la dernière échoue. Pour de plus amples informations et d'exemples, consultez Gestion des codes de sortie dans les commandes Fonctionnalité Exécuter la commande (p. 964).

Agent SSMne s'exécute pas correctement

Si vous rencontrez des problèmes lors de l'exécution des commandes avec Fonctionnalité Exécuter la commande , cela peut venir d' Agent SSM . Pour plus d'informations sur l'examen des problèmes liés à Agent SSM , consultez Dépannage de Agent SSM (p. 115) .

AWS Systems Manager State Manager

> **Note**
>
> State Manager et Fenêtres de maintenance peuvent effectuer des types similaires de mise à jour sur vos instances gérées. Votre choix dépend de la nécessité d'automatiser la conformité du système ou d'effectuer des tâches prioritaires et urgentes uniquement pendant les périodes que vous spécifiez.
> Pour plus d'informations, consultez Choisir entre State Manager etFenêtres de maintenance (p. 1405).

AWS Systems Manager State Manager est un service de gestion de configuration sécurisé et scalable qui automatise le processus de conservation de votre infrastructure Amazon EC2 et hybride à un état que vous définissez.

La liste suivante décrit les types de tâches que vous pouvez effectuer avec State Manager.

- Amorcer les instances avec un logiciel spécifique au démarrage
- Télécharger et mettre à jour les agents selon un programme défini, y compris Agent SSM
- Configurer les paramètres réseau
- Joindre des instances à un domaine Windows (instances Windows Server uniquement).
- Corriger les instances avec les mises à jour logiciel tout au long de leur cycle de vie
- Exécuter les scripts sur les instance gérées Linux et Windows tout au long de leur cycle de vie

State Manager s'intègre à AWS CloudTrail pour fournir un enregistrement de toutes les exécutions pour que vous puissiez réaliser un audit, ainsi qu'à Amazon EventBridge pour suivre les changements d'état. Vous pouvez également choisir de stocker et d'afficher la sortie de commande détaillée dans Amazon S3. Pour de plus amples informations, consultez les rubriques suivantes :

- Journalisation des appels d'API AWS Systems Manager avec AWS CloudTrail (p. 1336)
- Surveillance des événements Systems Manager avec Amazon EventBridge (p. 1340)
- (Facultatif) Configurer les intégrations avec d'autres services AWS (p. 47)

Prise en charge de Amazon EventBridge

Cette fonction Systems Manager est prise en charge en tant que type d'événement dans les règles EventBridge. Pour plus d'informations, consultez Surveillance des événements Systems Manager avec Amazon EventBridge (p. 1340) et Référence : modèles et types Amazon EventBridge d'événements pourSystems Manager (p. 1387).

Démarrez avec State Manager

Pour commencer à utiliser State Manager, procédez comme suit :

Tâche	Pour en savoir plus
Vérifiez les conditions préalables de Systems Manager	Prérequis d'Systems Manager (p. 15)
En savoir plus sur State Manager	À propos d'State Manager (p. 988)
Créez et attribuez une association State Manager à vos instances	Créer une association (p. 993)

Contenu connexe

- Combattre l'écart de configuration à l'aide d' Amazon EC2 Systems Manager et PowerShell de Windows DSC
- Configurer des Amazon EC2 instances dans un groupe Auto Scaling avec State Manager

Rubriques

- À propos d'State Manager (p. 988)
- Utilisation d'associations dans Systems Manager (p. 989)
- Procédures State Manager AWS Systems Manager (p. 1019)

À propos d'State Manager

AWS Systems ManagerState Managerest un service sécurisé et scalable qui automatise le processus de conservation de votre infrastructure Amazon Elastic Compute Cloud (Amazon EC2) et hybride à un état que vous définissez.

Voici comment State Manager fonctionne :

1. Déterminez l'état que vous souhaitez appliquer à vos instances gérées.

   Souhaitez-vous garantir que votre instance gérée est configurée avec des applications spécifiques, comme des anti-virus ou des applications anti-programme malveillant ? Souhaitez-vous automatiser le processus de mise à jour de l' Agent SSM ou d'autres AWS packages tels que AWSPVDriver ? Devez-vous vous assurer que des ports spécifiques sont fermés ou ouverts ? Pour démarrer avec State Manager, déterminez l'état que vous souhaitez appliquer à vos instances gérées. L'état que vous souhaitez appliquer détermine le document SSM que vous utilisez pour créer une association State Manager.

   Une association State Manager est une configuration qui est attribuée à vos instances gérées. La configuration définit l'état que vous souhaitez conserver sur vos instances. Par exemple, une association peut spécifier qu'un logiciel antivirus doit être installé et s'exécuter sur vos instances, ou que certains ports doivent être fermés. L'association spécifie un programme pour le moment où la configuration est appliquée une fois ou réappliquée à des heures spécifiées. L'association spécifie également des actions à exécuter lors de l'application de la configuration. Par exemple, une association pour un logiciel antivirus peut s'exécuter une fois par jour. Si le logiciel n'est pas installé, State Manager procède à son installation. Si le logiciel est installé, mais que le service ne s'exécute pas, l'association peut demander à State Manager de démarrer le service.
2. Déterminez si un document SSM préconfiguré peut vous aider à créer l'association State Manager.

   State Manager utilise des documents SSM pour créer une association. Systems Manager inclut des dizaines de SSM documents préconfigurés que vous pouvez utiliser pour créer une association. Les documents préconfigurés sont prêts à effectuer des tâches courantes, telles que l'installation d'applications, la configuration, l'exécution d'Amazon CloudWatchautomatisations, l'exécution Systems Manager et de scripts Shell, et la jonction d'un domaine Directory Service pour Active Directory, pour n'en citer que quelques-unes.PowerShell Il vous suffit de spécifier le nom du document et des informations sur les paramètres obligatoires, puis d'exécuter la commande pour créer l'association.

   Vous pouvez afficher tous les documents SSM dans la console Systems Manager. Choisissez le nom d'un document pour en savoir plus sur celui-ci. Voici deux exemples : AWS-ConfigureAWSPackage et AWS-InstallApplication.
3. Créez l'association.

   Vous pouvez créer l'association à l'aide de la AWS Systems Manager console , de l' AWS Command Line Interface (AWS CLI), de lOutils AWS pour Windows PowerShell' ou de l'API .Systems Manager Lorsque vous créez l'association, vous spécifiez les informations suivantes :

- Les paramètres pour le document de commande SSM (par exemple, le chemin d'accès à l'application à installer ou le script à exécuter sur les instances).
- Des cibles pour l'association. Vous pouvez cibler les instances gérées en spécifiant Amazon EC2 des balises, en choisissant une instance individuelle ou en choisissant un groupe dans IDs .Groupes de ressources AWS Vous pouvez également cibler toutes les instances gérées dans la région AWS et le compte AWS actuels.
- Un programme définissant quand et à quelle fréquence appliquer l'état. Vous pouvez spécifier une expression cron ou de fréquence. Pour plus d'informations sur la création de programmes à l'aide d'expressions cron et de fréquence (rate), consultez Expressions cron et rate pour les associations (p. 1397).

Lorsque vous exécutez la commande pour créer l'association, Systems Manager lie les informations que vous avez spécifiées (programme, cibles, documents SSM et paramètres) aux instances gérées. Le statut de l'association affiche initialement « Pending » (En suspens) pendant que système tente d'atteindre toutes les cibles et d'appliquer immédiatement l'état spécifié dans l'association.

> **Note**
>
> Si vous avez créé une nouvelle association qui est prévue pendant qu'une association précédente est encore en cours d'exécution, l'association précédente prend fin et la nouvelle association est exécutée.

Systems Manager signale le statut de la demande de création d'associations sur les instances gérées. Vous pouvez afficher les détails du statut dans la console ou à l'aide de l'action DescribeInstanceAssociationsStatus d'API . Si vous choisissez d'écrire la sortie de la commande dans Amazon Simple Storage Service (Amazon S3) lorsque vous créez une association, vous pouvez également afficher la sortie dans le Amazon S3 compartiment que vous spécifiez.

Pour plus d'informations, consultez la section Créer une association (p. 993).

4. Surveillez et mettez à jour.

   Une fois que vous avez créé l'association, State Manager réapplique la configuration selon le programme que vous avez défini dans l'association. Vous pouvez afficher le statut de vos associations sur la page State Manager dans la console ou en appelant directement l'ID d'association généré par Systems Manager lors de la création de l'association. Pour plus d'informations, consultez la section Affichage des historiques des associations (p. 1012). Vous pouvez mettre à jour vos documents d'association et les réappliquer si nécessaire. Vous pouvez également créer plusieurs versions d'une association. Pour plus d'informations, consultez la section Modifier et créer la nouvelle version d'une association (p. 1000).

Utilisation d'associations dans Systems Manager

Une association State Manager est une configuration qui est attribuée à vos instances gérées. La configuration définit l'état que vous souhaitez conserver sur vos instances. Par exemple, une association peut spécifier qu'un logiciel antivirus doit être installé et s'exécuter sur vos instances, ou que certains ports doivent être fermés. L'association spécifie un programme pour le moment où la configuration est appliquée une fois ou réappliquée à des heures spécifiées. L'association spécifie également des actions à exécuter lors de l'application de la configuration. Par exemple, une association pour un logiciel antivirus peut s'exécuter une fois par jour. Si le logiciel n'est pas installé, State Manager procède à son installation. Si le logiciel est installé, mais que le service ne s'exécute pas, l'association peut demander à State Manager de démarrer le service.

Les rubriques suivantes vous aideront à créer et à gérer des associations State Manager.

Rubriques

- À propos des cibles et des contrôles du débit dans les associations State Manager (p. 990)

- Créer une association (p. 993)
- Modifier et créer la nouvelle version d'une association (p. 1000)
- Affichage des historiques des associations (p. 1012)

À propos des cibles et des contrôles du débit dans les associations State Manager

Cette rubrique décrit les fonctionnalités State Manager qui vous aident à déployer une association sur des dizaines ou des centaines d'instances, tout en contrôlant le nombre d'instances qui exécutent l'association à l'heure prévue.

Targets

Lorsque vous créez une association State Manager, vous choisissez les instances à configurer avec cette dernière dans la section Targets (Cibles) de la console Systems Manager, comme illustré ici.

Si vous créez une association à l'aide d'un outil de ligne de commande tel que l'AWS CLI, vous spécifiez le paramètre `targets`. Le ciblage des instances vous permet de configurer des dizaines, des centaines ou des milliers d'instances avec une association sans devoir spécifier ou choisir une instance individuelleIDs.

> **Note**
>
> Vous pouvez uniquement cibler les instances Systems Manager gérées. Cela signifie que vous devez définir et configurer vos instances Systems Manager avant de créer une association. Pour plus d'informations, consultez la section Configuration d'AWS Systems Manager (p. 28).

La fonctionnalité State Manager inclut les options cibles suivantes lors de la création d'une association.

Spécification de balises d'instance

Utilisez cette option pour spécifier une clé de balise et (éventuellement) une valeur de balise actuellement affectée à vos instances. Lorsque vous exécutez la demande, le système recherche et tente de créer l'association sur toutes les instances qui correspondent à la clé et à la valeur de balise spécifiées. Lorsque le système crée l'association, il l'exécute. Après cette première exécution, le système exécute l'association selon la planification que vous avez spécifiée.

Si vous créez de nouvelles instances et affectez la clé de balise et la valeur spécifiées à ces dernières, le système applique automatiquement l'association, l'exécute immédiatement, puis l'exécute conformément

à la planification. Si vous supprimez les balises spécifiées d'une instance, le système n'exécute plus l'association sur ces instances.

Pour de plus amples informations sur l'affectation de balises à vos instances, consultez Balisage de vos ressources Amazon EC2 dans le Manuel de l'utilisateur Amazon EC2.

Choix manuel des instances

Utilisez cette option pour sélectionner manuellement les instances dans lesquelles vous souhaitez créer l'association. Le volet Instances affiche toutes les instances Systems Manager gérées dans le compte et la région AWS actuels. Vous pouvez sélectionner manuellement autant d'instances que vous le souhaitez. Lorsque le système crée l'association, il l'exécute. Après cette première exécution, le système exécute l'association selon la planification que vous avez spécifiée.

> **Note**
>
> Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.

Pour choisir un groupe de ressources

Utilisez cette option pour créer une association sur toutes les instances renvoyées par une requête basée sur une balise Groupes de ressources AWS ou une pile AWS CloudFormation.

Notez les détails suivants sur le ciblage des groupes de ressources pour une association.

- Si vous ajoutez de nouvelles instances à un groupe, le système mappe automatiquement les instances à l'association qui cible le groupe de ressources. Le système applique l'association aux instances lorsqu'il détecte la modification. Après cette première exécution, le système exécute l'association selon la planification que vous avez spécifiée.
- Si vous supprimez un groupe de ressources, toutes les instances de ce groupe n'exécutent plus l'association. Il est recommandé de supprimer les associations ciblant le groupe.
- Vous ne pouvez cibler qu'un seul groupe de ressources pour une association. Les groupes multiples ou imbriqués ne sont pas pris en charge.
- Après avoir créé une association, State Manager met à jour périodiquement l'association avec des informations sur les ressources du groupe de ressources. Si vous ajoutez de nouvelles ressources à un groupe de ressources, la planification du moment où le système applique l'association aux nouvelles ressources dépend de plusieurs facteurs. Vous pouvez vérifier l'état de l'association dans la page State Manager de la console Systems Manager.

> **Warning**
>
> Un utilisateur, un groupe ou un rôle AWS Identity and Access Management (IAM) autorisé à créer une association qui cible un groupe de ressources d'instances Amazon EC2 dispose automatiquement d'un contrôle au niveau racine de toutes les instances du groupe. Seuls les administrateurs approuvés doivent être autorisés à créer des associations.

Pour de plus amples informations sur Resource Groups, consultez Que sont les groupes de ressources AWS ? dans le Manuel de l'utilisateur Groupes de ressources AWS.

Choisir toutes les instances

Utilisez cette option pour cibler toutes les instances du compte et de la région AWS actuels. Lorsque vous exécutez la demande, le système localise et tente de créer l'association sur toutes les instances du compte et de la région AWS actuels. Lorsque le système crée l'association, il l'exécute. Après cette première exécution, le système exécute l'association selon la planification que vous avez spécifiée. Si vous créez de nouvelles instances, le système applique automatiquement l'association, l'exécute immédiatement, puis l'exécute conformément à la planification.

Contrôles du débit

Vous pouvez contrôler l'exécution d'une association sur vos instances en spécifiant une valeur de simultanéité et un seuil d'erreur. La valeur de simultanéité spécifie le nombre d'instances autorisées à exécuter l'association simultanément. Un seuil d'erreur spécifie le nombre d'exécutions d'associations qui peuvent échouer avant que Systems Manager n'envoie une commande à chaque instance configurée avec cette association pour qu'elle arrête de l'exécuter. La commande arrête l'exécution de l'association jusqu'à la prochaine exécution planifiée. Les fonctions de simultanéité et de seuil d'erreurs sont appelées collectivement contrôles du débit.

Concurrency

La simultanéité permet de limiter l'impact sur vos instances en vous permettant de spécifier que seul un certain nombre d'instances peuvent traiter une association à la fois. Vous pouvez spécifier un nombre absolu d'instances, par exemple 20, ou un pourcentage de l'ensemble d'instances cible, par exemple 10 %.

La simultanéité State Manager comporte les restrictions et limitations suivantes :

- Si vous choisissez de créer une association à l'aide de cibles, mais que vous ne spécifiez pas de valeur de simultanéité, State Manager applique automatiquement une simultanéité maximale de 50 instances.
- Si de nouvelles instances qui correspondent aux critères cibles sont mises en ligne pendant qu'une association qui utilise la simultanéité s'exécute, les nouvelles instances exécutent l'association si la valeur de simultanéité n'est pas dépassée. Si la valeur de simultanéité est dépassée, les instances sont ignorées pendant l'intervalle d'exécution d'association actuel. Les instances exécutent l'association lors du prochain intervalle programmé tout en se conformant aux exigences de simultanéité.
- Si vous mettez à jour une association qui utilise la simultanéité, et qu'une ou plusieurs instances traitent cette association lorsque celle-ci est mise à jour, les instances qui exécutent l'association sont autorisées à se terminer. Les associations qui n'ont pas commencé sont abandonnées. Une fois que les associations en cours d'exécution sont terminées, toutes les instances cibles ré-exécutent immédiatement l'association, car celle-ci a été mise à jour. Lorsque l'association s'exécute à nouveau, la valeur de simultanéité est appliquée.

Seuils d'erreurs

Un seuil d'erreur spécifie le nombre d'exécutions d'associations qui peuvent échouer avant que Systems Manager n'envoie une commande à chaque instance configurée avec cette association. La commande arrête l'exécution de l'association jusqu'à la prochaine exécution planifiée. Vous pouvez spécifier un nombre absolu d'erreurs, par exemple 10, ou un pourcentage de l'ensemble de la cible, par exemple 10 %.

Par exemple, si vous spécifiez un nombre absolu de trois erreurs, State Manager envoie la commande d'arrêt lorsque la quatrième erreur est renvoyée. Si vous spécifiez 0, State Manager envoie la commande d'arrêt dès que le premier résultat d'erreur est renvoyé.

Si vous spécifiez un seuil d'erreurs de 10 % pour 50 associations, State Manager envoie la commande d'arrêt lorsque la sixième erreur est renvoyée. Les associations qui s'exécutent déjà quand un seuil d'erreurs est atteint sont autorisées à se terminer, mais certaines de ces associations peuvent échouer. Pour vous assurer qu'il n'y pas plus d'erreurs que le nombre spécifié pour le seuil d'erreurs, définissez la valeur de Concurrency (Simultanéité) sur 1 afin que les associations s'exécutent une par une.

Les seuils d'erreurs State Manager comportent les limites et restrictions suivantes :

- Les seuils d'erreurs sont appliqués pour l'intervalle actuel.
- Des informations sur chaque erreur, y compris des détails au niveau des étapes, sont enregistrées dans l'historique d'association.
- Si vous choisissez de créer une association à l'aide de cibles, mais que vous ne spécifiez pas de seuil d'erreurs, State Manager applique automatiquement un seuil de 100 échecs.

Créer une association

Les procédures suivantes décrivent comment créer une association State Manager à l'aide de la console AWS Systems Manager, de l'AWS Command Line Interface (AWS CLI) et d'Outils AWS pour Windows PowerShell.

> **Important**
>
> Les procédures suivantes décrivent comment créer une association qui utilise une `Command` ou un document `Policy`. Pour plus d'informations sur la création d'une association qui utilise un document `Automation`, consultez Exécution d'automatisations avec des déclencheurs à l'aide deState Manager (p. 349).

Lorsque vous créez une association State Manager, par défaut, le système l'exécute immédiatement sur les instances ou cibles spécifiées. Après l'exécution initiale, l'association s'exécute à des intervalles selon le programme que vous avez défini et les règles suivantes :

- Les associations sont exécutées uniquement sur les instances en ligne lorsque l'intervalle commence. Les instances hors connexion sont ignorées.
- State Manager tente d'exécuter l'association sur toutes les instances spécifiées ou ciblées au cours d'un intervalle.
- Si une association n'est pas exécutée au cours d'un intervalle (par exemple, parce qu'une valeur de simultanéité a limité par le nombre d'instances pouvant traiter l'association simultanément), State Manager tente d'exécuter l'association lors du prochain intervalle.
- State Manager enregistre un historique pour tous les intervalles ignorés. Vous pouvez afficher l'historique dans l'onglet Execution History (Historique d'exécution).

> **Note**
>
> Si vous ne souhaitez pas qu'une association s'exécute immédiatement après sa création, vous pouvez choisir l'option Appliquer l'association uniquement à l'intervalle Cron spécifié suivant dans la console Systems Manager.

La procédure suivante décrit comment utiliser des cibles et des contrôles de débit lors de la création d'une association. Pour de plus amples informations sur l'utilisation de ces modèles, consultez À propos des cibles et des contrôles du débit dans les associations State Manager (p. 990).

> **Warning**
>
> Un utilisateur, un groupe ou un rôle AWS Identity and Access Management (IAM) autorisé à créer une association qui cible un groupe de ressources d'instances Amazon Elastic Compute Cloud (Amazon EC2) dispose automatiquement d'un contrôle au niveau racine de toutes les instances du groupe. Seuls les administrateurs approuvés doivent être autorisés à créer des associations.

Créer une association (console)

La procédure suivante décrit comment utiliser la console Systems Manager pour créer une association State Manager.

Pour créer une association State Manager

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez State Manager, puis Créer une association.
3. Dans le champ Nom, spécifiez un nom. Cette action est facultative, mais recommandée. Un nom vous aide à mémoriser l'objectif de l'association. À cet effet, vous pouvez, par exemple, spécifier Mettre_à_jour_automatiquement_AWSPVDrivers_sur__instances_us us-west-2 pour une association. Les espaces ne sont pas autorisés dans le nom.
4. Dans la liste Document, choisissez l'option en regard du nom d'un document. Notez le type de document. Cette procédure s'applique uniquement aux documents `Policy` et `Command`. Pour plus d'informations sur la création d'une association qui utilise un document `Automation`, consultez Exécution d'automatisations avec des déclencheurs à l'aide deState Manager (p. 349).
5. Pour Parameters (Paramètres), spécifiez les paramètres d'entrée requis.
6. Pour Targets (Cibles), choisissez une option. Pour plus d'informations sur l'utilisation des cibles, consultez À propos des cibles et des contrôles du débit dans les associations State Manager (p. 990).
7. Dans la section Specify schedule (Spécifier le programme), choisissez On Schedule (Selon le calendrier) ou No schedule (Pas de calendrier). Si vous choisissez On Schedule (Selon planification), utilisez les boutons fournis pour créer une planification de type cron ou rate pour l'association.

   Si vous ne souhaitez pas que l'association s'exécute immédiatement après sa création, choisissez Appliquer l'association uniquement à l'intervalle Cron spécifié suivant.
8. Dans la section Options avancées, utilisez Conformité pour choisir un niveau de gravité pour l'association. Les rapports de conformité indiquent si l'état de l'association est conforme ou non conforme, ainsi que le niveau de gravité que vous spécifiez ici. Pour plus d'informations, consultez la section A propos de la conformité des associations State Manager (p. 783).
9. Dans la section Contrôle du débit, choisissez des options pour contrôler la façon dont l'association s'exécute sur plusieurs instances. Pour de plus amples informations sur l'utilisation des contrôles de débit, consultez À propos des cibles et des contrôles du débit dans les associations State Manager (p. 990).

   Dans la section Simultanéité, choisissez une option :

   - Choisissez targets (cibles) pour entrer un nombre absolu de cibles pouvant exécuter l'association simultanément.
   - Choisissez percentage (pourcentage) pour saisir un pourcentage de l'ensemble de cibles pouvant exécuter l'association simultanément.

   Dans la section Error threshold (Seuil d'erreurs), choisissez une option :

   - Choisissez errors (erreurs) pour saisir un nombre absolu d'erreurs autorisées avant que State Manager ne cesse d'exécuter des associations sur des cibles supplémentaires.
   - Choisissez percentage (pourcentage) pour saisir un pourcentage d'erreurs autorisées avant que State Manager ne cesse d'exécuter des associations sur des cibles supplémentaires.
10. (Facultatif) Dans Output options (Options de sortie), pour enregistrer la sortie de la commande dans un fichier, sélectionnez Enable writing to an S3 bucket (Autoriser l'écriture dans un compartiment S3) Entrez les noms de compartiment et de préfixe (dossier) dans les zones.

Note

Les autorisations S3 qui accordent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour de plus amples informations, veuillez consulter Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). En outre, si le compartiment S3 spécifié se trouve dans un compte AWS différent, assurez-vous que le profil d'instance associé à l'instance dispose des autorisations nécessaires pour écrire dans ce compartiment.

11. Choisissez Create Association (Créer une association).

Note

Si vous supprimez l'association que vous avez créée, celle-ci ne s'exécute plus sur aucune cible de cette association.

Créer une association (ligne de commande)

La procédure suivante décrit comment utiliser l'AWS CLI (sous Linux ou Windows) ou Outils AWS pour PowerShell pour créer une association State Manager. Cette section présente plusieurs exemples qui montrent comment utiliser les cibles et les contrôles du débit. Les cibles et les contrôles du débit vous permettent d'affecter une association à des dizaines ou des centaines d'instances tout en contrôlant la façon dont ces associations s'exécutent. Pour de plus amples informations sur les cibles et les contrôles du débit, consultez À propos des cibles et des contrôles du débit dans les associations State Manager (p. 990).

Avant de commencer

Le paramètre `targets` est un ensemble de critères de recherche qui cible les instances à l'aide d'une combinaison `Key`, `Value` que vous spécifiez. Si vous prévoyez de créer une association sur des dizaines ou des centaines d'instances à l'aide du paramètre `targets`, passez en revue les options de ciblage suivantes avant de commencer la procédure.

Cibler quelques instances en spécifiantIDs

```
--targets Key=InstanceIds,Values=instance-id-1,instance-id-2,instance-id-3
```

```
--targets
 Key=InstanceIds,Values=i-02573cafcfEXAMPLE,i-0471e04240EXAMPLE,i-07782c72faEXAMPLE
```

Cibler les instances à l'aide de balises Amazon EC2

```
--targets Key=tag:tag-key,Values=tag-value-1,tag-value-2,tag-value-3
```

```
--targets Key=tag:Environment,Values=Development,Test,Pre-production
```

Cibler les instances avec Groupes de ressources AWS

```
--targets Key=resource-groups:Name,Values=resource-group-name
```

```
--targets Key=resource-groups:Name,Values=WindowsInstancesGroup
```

Cibler toutes les instances dans le AWS compte et la AWS région actuels

```
--targets Key=InstanceIds,Values=*
```

Note

Lorsque vous créez une association, vous spécifiez à quel moment le programme s'exécute. Vous devez spécifier le programme à l'aide d'une expression de type cron ou rate. Pour plus d'informations sur les expressions de type cron et rate, consultez Expressions cron et rate pour les associations (p. 1397).

Créer une association

1. Installez et configurez l'AWS CLI ou Outils AWS pour PowerShell si vous ne l'avez pas déjà fait.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).
2. Utilisez le format suivant pour créer une commande qui crée une association State Manager.

   Linux

```
aws ssm create-association \
  --targets target_options \
  --name document_name \
  --schedule "cron_or_rate_expression" \
  --parameters (if any) \
  --max-concurrency a_number_of_instances_or_a_percentage_of_target_set \
  --max-errors a_number_of_errors_or_a_percentage_of_target_set
```

   Windows

```
aws ssm create-association ^
  --targets target_options ^
  --name document_name ^
  --schedule "cron_or_rate_expression" ^
  --parameters (if any) ^
  --max-concurrency a_number_of_instances_or_a_percentage_of_target_set ^
  --max-errors a_number_of_errors_or_a_percentage_of_target_set
```

   PowerShell

```
New-SSMAssociation `
  -AssociationName document_name `
  -Target target_options `
  -ScheduleExpression "cron_or_rate_expression" `
  -Parameters (if any) `
  -MaxConcurrency a_number_of_instances_or_a_percentage_of_target_set `
  -MaxError  a_number_of_errors_or_a_percentage_of_target_set
```

   L'exemple suivant crée une association sur les instances balisées avec `"Environment,Linux"` . L'association utilise le document AWS-UpdateSSMAgent pour mettre à jour Agent SSM les instances ciblées tous les dimanches à 2 h 00. Cette association s'exécute simultanément sur 10 instances maximum à un moment donné. En outre, cette association cesse de s'exécuter sur d'autres instances pour un intervalle d'exécution particulier si le nombre d'erreurs dépasse 5. Pour les rapports de conformité, cette association se voit attribuer un niveau de gravité Medium (Moyenne).

   Linux

```
aws ssm create-association \
  --association-name Update_SSM_Agent_Linux \
  --targets Key=tag:Environment,Values=Linux \
  --name AWS-UpdateSSMAgent  \
```

```
  --compliance-severity "MEDIUM" \
  --schedule "cron(0 2 ? * SUN *)" \
  --max-errors "5" \
  --max-concurrency "10"
```

Windows

```
aws ssm create-association ^
  --association-name Update_SSM_Agent_Linux ^
  --targets Key=tag:Environment,Values=Linux ^
  --name AWS-UpdateSSMAgent  ^
  --compliance-severity "MEDIUM" ^
  --schedule "cron(0 2 ? * SUN *)" ^
  --max-errors "5" ^
  --max-concurrency "10"
```

PowerShell

```
New-SSMAssociation `
  -AssociationName Update_SSM_Agent_Linux `
  -Name AWS-UpdateSSMAgent `
  -Target @{
      "Key"="tag:Environment"
      "Values"="Linux"
    } `
  -ScheduleExpression "cron(0 2 ? * SUN *)" `
  -MaxConcurrency 10 `
  -MaxError 5 `
  -ComplianceSeverity MEDIUM
```

Les commandes suivantes créent une association qui analyse les instances pour rechercher les mises à jour de correctifs manquantes à l'aide du document AWS-RunPatchBaseline. Cette association cible toutes les instances gérées du compte dans la région us-east-2. L'association spécifie les `Operation` paramètres `RebootOption` et .

Linux

```
aws ssm create-association \
--name "AWS-RunPatchBaseline" \
--association-name "ScanningInstancesForMissingUpdate" \
--targets "Key=instanceids,Values=*" \
--parameters "Operation=Scan,RebootOption=NoReboot" \
--region us-east-2
```

Windows

```
aws ssm create-association ^
--name "AWS-RunPatchBaseline" ^
--association-name "ScanningInstancesForMissingUpdate" ^
--targets "Key=instanceids,Values=*" ^
--parameters "Operation=Scan,RebootOption=NoReboot" ^
--region us-east-2
```

PowerShell

```
New-SSMAssociation `
-AssociationName ScanningInstancesForMissingUpdate `
-Name AWS-RunPatchBaseline `
```

```
-Target @{
      "Key"="instanceids"
      "Values"="*"
    } `
-Parameters "Operation=Scan,RebootOption=NoReboot" `
-Region us-east-2
```

L'exemple suivant cible l'instance IDs en spécifiant une valeur générique (*). Cela permet Systems Manager à de créer une association sur toutes les instances dans le compte et la région actuels. Cette association s'exécute simultanément sur 10 instances maximum à un moment donné. En outre, cette association cesse de s'exécuter sur d'autres instances pour un intervalle particulier si le nombre d'erreurs dépasse 5. Pour les rapports de conformité, cette association se voit attribuer un niveau de gravité Medium (Moyenne). Cette association s'exécute uniquement selon la planification Cron spécifiée. Il ne s'exécute pas immédiatement après la création de l'association.

Linux

```
aws ssm create-association \
  --association-name Update_SSM_Agent_Linux \
  --name "AWS-UpdateSSMAgent" \
  --targets "Key=instanceids,Values=*" \
  --compliance-severity "MEDIUM" \
  --schedule "cron(0 2 ? * SUN *)" \
  --max-errors "5" \
  --max-concurrency "10" \
  --apply-only-at-cron-interval
```

Windows

```
aws ssm create-association ^
      --association-name Update_SSM_Agent_Linux ^
      --name "AWS-UpdateSSMAgent" ^
      --targets "Key=instanceids,Values=*" ^
      --compliance-severity "MEDIUM" ^
      --schedule "cron(0 2 ? * SUN *)" ^
      --max-errors "5" ^
      --max-concurrency "10" ^
      --apply-only-at-cron-interval
```

PowerShell

```
New-SSMAssociation `
  -AssociationName Update_SSM_Agent_All `
  -Name AWS-UpdateSSMAgent `
  -Target @{
      "Key"="InstanceIds"
      "Values"="*"
    } `
  -ScheduleExpression "cron(0 2 ? * SUN *)" `
  -MaxConcurrency 10 `
  -MaxError 5 `
  -ComplianceSeverity MEDIUM `
  -ApplyOnlyAtCronInterval
```

L'exemple suivant crée une association sur des instances dans Groupes de ressources AWS. Le groupe est nommé « HR-Department ». L'association utilise le document AWS-UpdateSSMAgent pour mettre à jour Agent SSM les instances ciblées tous les dimanches à 2 h 00. Cette association s'exécute simultanément sur 10 instances maximum à un moment donné. En outre, cette association

arrête d'être exécutée sur des instances supplémentaires pour un intervalle d'exécution particulier si le nombre d'erreurs dépasse 5. Pour les rapports de conformité, cette association se voit attribuer un niveau de gravité Medium (Moyenne). Cette association s'exécute uniquement selon la planification Cron spécifiée. Il ne s'exécute pas immédiatement après la création de l'association.

Linux

```
aws ssm create-association \
  --association-name Update_SSM_Agent_Linux \
  --targets Key=resource-groups:Name,Values=HR-Department \
  --name AWS-UpdateSSMAgent  \
  --compliance-severity "MEDIUM" \
  --schedule "cron(0 2 ? * SUN *)" \
  --max-errors "5" \
  --max-concurrency "10" \
  --apply-only-at-cron-interval
```

Windows

```
aws ssm create-association ^
  --association-name Update_SSM_Agent_Linux ^
  --targets Key=resource-groups:Name,Values=HR-Department ^
  --name AWS-UpdateSSMAgent  ^
  --compliance-severity "MEDIUM" ^
  --schedule "cron(0 2 ? * SUN *)" ^
  --max-errors "5" ^
  --max-concurrency "10" ^
  --apply-only-at-cron-interval
```

PowerShell

```
New-SSMAssociation `
  -AssociationName Update_SSM_Agent_Linux `
  -Name AWS-UpdateSSMAgent `
  -Target @{
      "Key"="resource-groups:Name"
      "Values"="HR-Department"
    } `
  -ScheduleExpression "cron(0 2 ? * SUN *)" `
  -MaxConcurrency 10 `
  -MaxError 5 `
  -ComplianceSeverity MEDIUM `
  -ApplyOnlyAtCronInterval
```

L'exemple suivant crée une association qui s'exécute sur des instances balisées avec un ID d'instance spécifique. L'association utilise le Agent SSM document pour mettre à jour l'agent SSM sur les instances ciblées une seule fois, à 15 h 55, heure UTC, le 7 juillet 2020. Cette association s'exécute simultanément sur 10 instances maximum à un moment donné. En outre, cette association arrête d'être exécutée sur des instances supplémentaires pour un intervalle d'exécution particulier si le nombre d'erreurs dépasse 5. Pour les rapports de conformité, cette association se voit attribuer un niveau de gravité Medium (Moyenne). Cette association s'exécute uniquement selon la planification Cron spécifiée. Il ne s'exécute pas immédiatement après la création de l'association.

Note

Si vous utilisez une `at()` expression, l'association ne sera exécutée qu'une seule fois. Elle ne sera pas appliquée aux nouvelles instances en ligne dans les balises ou le groupe de ressources de cette association. Les associations créées avec une date passée ou présente (dorénavant qu'elle soit traitée, la date est passée) s'exécutent immédiatement.

Linux

```
aws ssm create-association \
  --association-name One-Time-Association \
  --targets "Key=instanceids,Values=i-0cb2b964d3e14fd9f" \
  --name AWS-UpdateSSMAgent  \
  --compliance-severity "MEDIUM" \
  --schedule "at(2020-07-07T15:55:00)" \
  --max-errors "5" \
  --max-concurrency "10" \
  --apply-only-at-cron-interval
```

Windows

```
aws ssm create-association ^
  --association-name Update_SSM_Agent_Linux ^
  --targets "Key=instanceids,Values=i-0cb2b964d3e14fd9f" ^
  --name AWS-UpdateSSMAgent  ^
  --compliance-severity "MEDIUM" ^
  --schedule "at(2020-07-07T15:55:00)" ^
  --max-errors "5" ^
  --max-concurrency "10" ^
  --apply-only-at-cron-interval
```

PowerShell

```
New-SSMAssociation `
  -AssociationName Update_SSM_Agent_Linux `
  -Name AWS-UpdateSSMAgent `
  -Target @{
      "Key"="tag:instanceids"
      "Values"="i-0cb2b964d3e14fd9f"
    } `
  -ScheduleExpression "at(2020-07-07T15:55:00)" `
  -MaxConcurrency 10 `
  -MaxError 5 `
  -ComplianceSeverity MEDIUM `
  -ApplyOnlyAtCronInterval
```

Note

Si vous supprimez l'association que vous avez créée, celle-ci ne s'exécute plus sur aucune cible de cette association. En outre, si vous avez spécifié le paramètre `apply-only-at-cron-interval`, vous pouvez réinitialiser cette option. Pour ce faire, spécifiez le paramètre `no-apply-only-at-cron-interval` lorsque vous mettez à jour l'association à partir de la ligne de commande. Ce paramètre force l'association à s'exécuter immédiatement après la mise à jour de l'association et selon l'intervalle spécifié.

Modifier et créer la nouvelle version d'une association

Vous pouvez modifier une association pour spécifier un nouveau nom, un niveau de gravité ou des cibles. Vous pouvez également choisir d'écrire la sortie de la commande dans un compartiment Amazon Simple Storage Service (Amazon S3). Après avoir modifié une association, Systems Manager crée une nouvelle version. Vous pouvez afficher différentes versions après modification, comme décrit dans les procédures suivantes.

Les procédures suivantes expliquent comment modifier et créer une nouvelle version d'une association à l'aide de la console AWS Systems Manager, de l'AWS Command Line Interface (AWS CLI) et d'Outils AWS pour PowerShell.

Modifier une association (console)

La procédure suivante décrit comment utiliser la console Systems Manager pour modifier et créer une nouvelle version d'une association.

> **Note**
>
> Cette procédure requiert que vous possédiez un accès en écriture à un compartiment S3 existant. Si vous n'avez encore jamais utilisé Amazon S3, sachez que des frais s'appliquent l'utilisation de Amazon S3. Pour plus d'informations sur la création d'un compartiment, consultez Créer un compartiment.

Pour modifier une association State Manager

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez State Manager.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez State Manager.
3. Sélectionnez l'association que vous avez créée dans , puis choisissez

La procédure suivante décrit comment utiliser l'AWS CLI (sous Linux ou Windows) ou Outils AWS pour PowerShell pour créer une association State Manager. Cette section présente plusieurs exemples qui montrent comment utiliser les cibles et les contrôles du débit. Les cibles et les contrôles du débit vous permettent d'affecter une association à des dizaines ou des centaines d'instances tout en contrôlant la façon dont ces associations s'exécutent. Pour de plus amples informations sur les cibles et les contrôles du débit, consultez À propos des cibles et des contrôles du débit dans les associations State Manager (p. 990).

Avant de commencer

Le paramètre `targets` est un ensemble de critères de recherche qui cible les instances à l'aide d'une combinaison `Key`, `Value` que vous spécifiez. Si vous prévoyez de créer une association sur des dizaines ou des centaines d'instances à l'aide du paramètre `targets`, passez en revue les options de ciblage suivantes avant de commencer la procédure.

Cibler quelques instances en spécifiantIDs

```
--targets Key=InstanceIds,Values=instance-id-1,instance-id-2,instance-id-3
```

```
--targets
 Key=InstanceIds,Values=i-02573cafcfEXAMPLE,i-0471e04240EXAMPLE,i-07782c72faEXAMPLE
```

Cibler les instances à l'aide de balises Amazon EC2

```
--targets Key=tag:tag-key,Values=tag-value-1,tag-value-2,tag-value-3
```

```
--targets Key=tag:Environment,Values=Development,Test,Pre-production
```

Cibler les instances avec Groupes de ressources AWS

```
--targets Key=resource-groups:Name,Values=resource-group-name
```

```
--targets Key=resource-groups:Name,Values=WindowsInstancesGroup
```

Cibler toutes les instances dans le AWS compte et la AWS région actuels

```
--targets Key=InstanceIds,Values=*
```

> **Note**
>
> Lorsque vous créez une association, vous spécifiez à quel moment le programme s'exécute. Vous devez spécifier le programme à l'aide d'une expression de type cron ou rate. Pour plus d'informations sur les expressions de type cron et rate, consultez Expressions cron et rate pour les associations (p. 1397).

Créer une association

1. Installez et configurez l'AWS CLI ou Outils AWS pour PowerShell si vous ne l'avez pas déjà fait.
   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).
2. Utilisez le format suivant pour créer une commande qui crée une association State Manager.
   Linux

   ```
   aws ssm create-association \
     --targets target_options \
     --name document_name \
     --schedule "cron_or_rate_expression" \
     --parameters (if any) \
     --max-
   concurrency a_number_of_instances_or_a_percentage_of_target_set \
     --max-errors a_number_of_errors_or_a_percentage_of_target_set
   ```

   Windows

   ```
   aws ssm create-association ^
     --targets target_options ^
     --name document_name ^
     --schedule "cron_or_rate_expression" ^
     --parameters (if any) ^
     --max-
   concurrency a_number_of_instances_or_a_percentage_of_target_set ^
     --max-errors a_number_of_errors_or_a_percentage_of_target_set
   ```

   PowerShell

   ```
   New-SSMAssociation `
     -AssociationName document_name `
     -Target target_options `
     -ScheduleExpression "cron_or_rate_expression" `
     -Parameters (if any) `
   ```

```
  -
MaxConcurrency a_number_of_instances_or_a_percentage_of_target_set
 `
  -MaxError  a_number_of_errors_or_a_percentage_of_target_set
```

L'exemple suivant crée une association sur les instances balisées avec `"Environment,Linux"` . L'association utilise le document AWS-UpdateSSMAgent pour mettre à jour Agent SSM les instances ciblées tous les dimanches à 2 h 00. Cette association s'exécute simultanément sur 10 instances maximum à un moment donné. En outre, cette association cesse de s'exécuter sur d'autres instances pour un intervalle d'exécution particulier si le nombre d'erreurs dépasse 5. Pour les rapports de conformité, cette association se voit attribuer un niveau de gravité Medium (Moyenne).

Linux

```
aws ssm create-association \
  --association-name Update_SSM_Agent_Linux \
  --targets Key=tag:Environment,Values=Linux \
  --name AWS-UpdateSSMAgent  \
  --compliance-severity "MEDIUM" \
  --schedule "cron(0 2 ? * SUN *)" \
  --max-errors "5" \
  --max-concurrency "10"
```

Windows

```
aws ssm create-association ^
  --association-name Update_SSM_Agent_Linux ^
  --targets Key=tag:Environment,Values=Linux ^
  --name AWS-UpdateSSMAgent  ^
  --compliance-severity "MEDIUM" ^
  --schedule "cron(0 2 ? * SUN *)" ^
  --max-errors "5" ^
  --max-concurrency "10"
```

PowerShell

```
New-SSMAssociation `
  -AssociationName Update_SSM_Agent_Linux `
  -Name AWS-UpdateSSMAgent `
  -Target @{
      "Key"="tag:Environment"
      "Values"="Linux"
    } `
  -ScheduleExpression "cron(0 2 ? * SUN *)" `
  -MaxConcurrency 10 `
  -MaxError 5 `
  -ComplianceSeverity MEDIUM
```

Les commandes suivantes créent une association qui analyse les instances pour rechercher les mises à jour de correctifs manquantes à l'aide du document AWS-RunPatchBaseline. Cette association cible toutes les instances gérées du compte dans la région us-east-2. L'association spécifie les `Operation` paramètres `RebootOption` et .

Linux

```
aws ssm create-association \
--name "AWS-RunPatchBaseline" \
--association-name "ScanningInstancesForMissingUpdate" \
--targets "Key=instanceids,Values=*" \
--parameters "Operation=Scan,RebootOption=NoReboot" \
--region us-east-2
```

Windows

```
aws ssm create-association ^
--name "AWS-RunPatchBaseline" ^
--association-name "ScanningInstancesForMissingUpdate" ^
--targets "Key=instanceids,Values=*" ^
--parameters "Operation=Scan,RebootOption=NoReboot" ^
--region us-east-2
```

PowerShell

```
New-SSMAssociation `
-AssociationName ScanningInstancesForMissingUpdate `
-Name AWS-RunPatchBaseline `
-Target @{
      "Key"="instanceids"
      "Values"="*"
    } `
-Parameters "Operation=Scan,RebootOption=NoReboot" `
-Region us-east-2
```

L'exemple suivant cible l'instance IDs en spécifiant une valeur générique (*). Cela permet Systems Manager à de créer une association sur toutes les instances dans le compte et la région actuels. Cette association s'exécute simultanément sur 10 instances maximum à un moment donné. En outre, cette association cesse de s'exécuter sur d'autres instances pour un intervalle particulier si le nombre d'erreurs dépasse 5. Pour les rapports de conformité, cette association se voit attribuer un niveau de gravité Medium (Moyenne). Cette association s'exécute uniquement selon la planification Cron spécifiée. Il ne s'exécute pas immédiatement après la création de l'association.

Linux

```
aws ssm create-association \
  --association-name Update_SSM_Agent_Linux \
  --name "AWS-UpdateSSMAgent" \
  --targets "Key=instanceids,Values=*" \
  --compliance-severity "MEDIUM" \
  --schedule "cron(0 2 ? * SUN *)" \
  --max-errors "5" \
  --max-concurrency "10" \
  --apply-only-at-cron-interval
```

Windows

```
aws ssm create-association ^
      --association-name Update_SSM_Agent_Linux ^
      --name "AWS-UpdateSSMAgent" ^
```

```
      --targets "Key=instanceids,Values=*" ^
      --compliance-severity "MEDIUM" ^
      --schedule "cron(0 2 ? * SUN *)" ^
      --max-errors "5" ^
      --max-concurrency "10" ^
      --apply-only-at-cron-interval
```

PowerShell

```
New-SSMAssociation `
  -AssociationName Update_SSM_Agent_All `
  -Name AWS-UpdateSSMAgent `
  -Target @{
      "Key"="InstanceIds"
      "Values"="*"
    } `
  -ScheduleExpression "cron(0 2 ? * SUN *)" `
  -MaxConcurrency 10 `
  -MaxError 5 `
  -ComplianceSeverity MEDIUM `
  -ApplyOnlyAtCronInterval
```

L'exemple suivant crée une association sur des instances dans Groupes de ressources AWS. Le groupe est nommé « HR-Department ». L'association utilise le document AWS-UpdateSSMAgent pour mettre à jour Agent SSM les instances ciblées tous les dimanches à 2 h 00. Cette association s'exécute simultanément sur 10 instances maximum à un moment donné. En outre, cette association arrête d'être exécutée sur des instances supplémentaires pour un intervalle d'exécution particulier si le nombre d'erreurs dépasse 5. Pour les rapports de conformité, cette association se voit attribuer un niveau de gravité Medium (Moyenne). Cette association s'exécute uniquement selon la planification Cron spécifiée. Il ne s'exécute pas immédiatement après la création de l'association.

Linux

```
aws ssm create-association \
  --association-name Update_SSM_Agent_Linux \
  --targets Key=resource-groups:Name,Values=HR-Department \
  --name AWS-UpdateSSMAgent  \
  --compliance-severity "MEDIUM" \
  --schedule "cron(0 2 ? * SUN *)" \
  --max-errors "5" \
  --max-concurrency "10" \
  --apply-only-at-cron-interval
```

Windows

```
aws ssm create-association ^
  --association-name Update_SSM_Agent_Linux ^
  --targets Key=resource-groups:Name,Values=HR-Department ^
  --name AWS-UpdateSSMAgent  ^
  --compliance-severity "MEDIUM" ^
  --schedule "cron(0 2 ? * SUN *)" ^
  --max-errors "5" ^
  --max-concurrency "10" ^
  --apply-only-at-cron-interval
```

PowerShell

```
New-SSMAssociation `
```

```
  -AssociationName Update_SSM_Agent_Linux `
  -Name AWS-UpdateSSMAgent `
  -Target @{
      "Key"="resource-groups:Name"
      "Values"="HR-Department"
    } `
  -ScheduleExpression "cron(0 2 ? * SUN *)" `
  -MaxConcurrency 10 `
  -MaxError 5 `
  -ComplianceSeverity MEDIUM `
  -ApplyOnlyAtCronInterval
```

L'exemple suivant crée une association qui s'exécute sur des instances balisées avec un ID d'instance spécifique. L'association utilise le Agent SSM document pour mettre à jour l'agent SSM sur les instances ciblées une seule fois, à 15 h 55, heure UTC, le 7 juillet 2020. Cette association s'exécute simultanément sur 10 instances maximum à un moment donné. En outre, cette association arrête d'être exécutée sur des instances supplémentaires pour un intervalle d'exécution particulier si le nombre d'erreurs dépasse 5. Pour les rapports de conformité, cette association se voit attribuer un niveau de gravité Medium (Moyenne). Cette association s'exécute uniquement selon la planification Cron spécifiée. Il ne s'exécute pas immédiatement après la création de l'association.

Note

Si vous utilisez une `at()` expression, l'association ne sera exécutée qu'une seule fois. Elle ne sera pas appliquée aux nouvelles instances en ligne dans les balises ou le groupe de ressources de cette association. Les associations créées avec une date passée ou présente (dorénavant qu'elle soit traitée, la date est passée) s'exécutent immédiatement.

Linux

```
aws ssm create-association \
```

```
  --association-name One-Time-Association \
```

```
  --targets "Key=instanceids,Values=i-0cb2b964d3e14fd9f" \
```

```
  --name AWS-UpdateSSMAgent  \
```

```
  --compliance-severity "MEDIUM" \
```

```
  --schedule "at(2020-07-07T15:55:00)" \
```

```
  --max-errors "5" \
```

```
  --max-concurrency "10" \
```

```
  --apply-only-at-cron-interval
```

Windows

```
aws ssm create-association ^
```

```
  --association-name Update_SSM_Agent_Linux ^
```

```
  --targets "Key=instanceids,Values=i-0cb2b964d3e14fd9f" ^
```

```
  --name AWS-UpdateSSMAgent  ^
```

PowerShell

```
New-SSMAssociation `
  -AssociationName Update_SSM_Agent_Linux `
  -Name AWS-UpdateSSMAgent `
  -Target @{
      "Key"="tag:instanceids"
      "Values"="i-0cb2b964d3e14fd9f"
    } `
  -ScheduleExpression "at(2020-07-07T15:55:00)" `
  -MaxConcurrency 10 `
  -MaxError 5 `
  -ComplianceSeverity MEDIUM `
  -ApplyOnlyAtCronInterval
```

Note

Si vous supprimez l'association que vous avez créée, celle-ci ne s'exécute plus sur aucune cible de cette association. En outre, si vous avez spécifié le paramètre `apply-only-at-cron-interval`, vous pouvez réinitialiser cette option. Pour ce faire, spécifiez le paramètre `no-apply-only-at-cron-interval` lorsque vous mettez à jour l'association à partir de la ligne de commande. Ce paramètre force l'association à s'exécuter immédiatement après la mise à jour de l'association et selon l'intervalle spécifié.

(p. 995) Edit (Modifier).

4. Dans le champ Nom, entrez un nouveau nom. Par exemple, saisissez **`TestHostnameAssociation2`**.
5. Dans la section Spécifier le programme, choisissez une nouvelle option. Par exemple, choisissez Générateur de planification Cron, puis choisissez Toutes les heures.
6. (Facultatif) Dans Output options (Options de sortie), pour enregistrer la sortie de la commande dans un fichier, sélectionnez Enable writing to an S3 bucket (Autoriser l'écriture dans un compartiment S3) Entrez les noms de compartiment et de préfixe (dossier) dans les zones.

   Note

   Les autorisations S3 qui accordent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour de plus amples informations, veuillez consulter Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). En outre, si le compartiment S3 spécifié se trouve dans un compte AWS différent, assurez-vous que le profil d'instance associé à l'instance dispose des autorisations nécessaires pour écrire dans ce compartiment.
7. Choisissez Edit association.
8. Dans la page Associations, choisissez le nom de l'association que vous venez de modifier, puis choisissez l'onglet Versions. Le système liste chaque version de l'association que vous avez créée et modifiée.

9. Ouvrez la console Amazon S3 à l'adresse https://console.aws.amazon.com/s3/.
10. Choisissez le nom du compartiment S3 que vous avez spécifié pour le stockage de la sortie de commande, puis choisissez le dossier dont le nom est l'ID de l'instance qui a exécuté l'association. (Si vous avez choisi de stocker la sortie dans un dossier du compartiment, ouvrez-le en premier.)

Modifier une association (ligne de commande)

La procédure suivante décrit comment utiliser l'AWS CLI (sous Linux ou Windows) ou Outils AWS pour PowerShell pour modifier et créer une nouvelle version d'une association.

Pour modifier une association State Manager

1. Installez et configurez l'AWS CLI ou Outils AWS pour PowerShell si vous ne l'avez pas déjà fait.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Utilisez le format suivant pour créer une commande permettant de modifier et de créer une nouvelle version d'une association State Manager existante.

   Linux

```
aws ssm update-association \
  --association-id b85ccafe-9f02-4812-9b81-01234EXAMPLE \
  --association-name association_name \
  --parameters (if any) \
  --output-location S3Location='{OutputS3Region=region,OutputS3BucketName=DOC-
EXAMPLE-BUCKET,OutputS3KeyPrefix=keyprefix}' \
  --scheduleexpression "cron_or_rate_expression"
```

   > **Important**
   >
   > Pour conserver les valeurs de paramètre existantes pour votre association, telles que le nom de l'association ou le niveau de conformité, vous devez spécifier ces valeurs lors de la mise à jour de l'association. Si vous ne spécifiez pas ces valeurs de paramètre lors de la mise à jour d'une association, la nouvelle version d'association n'utilisera aucune valeur. Par exemple, si votre association existante possède une planification cron, mais que vous ne spécifiez pas `--schedule-expression` lors de la mise à jour, la nouvelle version d'association n'aura pas d'expression de planification.

   Windows

```
aws ssm update-association ^
  --association-id b85ccafe-9f02-4812-9b81-01234EXAMPLE ^
  --association-name association_name ^
  --parameters (if any) ^
  --output-location S3Location='{OutputS3Region=region,OutputS3BucketName=DOC-
EXAMPLE-BUCKET,OutputS3KeyPrefix=keyprefix}' ^
  --scheduleexpression "cron_or_rate_expression"
```

   > **Important**
   >
   > Pour conserver les valeurs de paramètre existantes pour votre association, telles que le nom de l'association ou le niveau de conformité, vous devez spécifier ces valeurs lors de la mise à jour de l'association. Si vous ne spécifiez pas ces paramètres lors de la mise à jour d'une association, la nouvelle version d'association utilisera les valeurs par défaut (none). Par exemple, si votre association existante possède une planification cron, mais que vous ne spécifiez pas `--schedule-expression` lors de la mise à jour, la nouvelle version d'association n'aura pas d'expression de planification.

   PowerShell

```
Update-SSMAssociation `
  -AssociationId b85ccafe-9f02-4812-9b81-01234EXAMPLE `
  -AssociationName document_name `
  -Parameter (if any) `
```

```
  -S3Location_OutputS3BucketName DOC-EXAMPLE-BUCKET `
  -S3Location_OutputS3KeyPrefix key_prefix `
  -S3Location_OutputS3Region region `
  -ScheduleExpression "cron_or_rate_expression"
```

> **Important**
>
> Pour conserver les valeurs de paramètre existantes pour votre association, telles que le nom de l'association ou le niveau de conformité, vous devez spécifier ces valeurs lors de la mise à jour de l'association. Si vous ne spécifiez pas ces paramètres lors de la mise à jour d'une association, la nouvelle version d'association n'utilisera aucune valeur. Par exemple, si votre association existante possède une planification cron, mais que vous ne spécifiez pas `-ScheduleExpression` lors de la mise à jour, la nouvelle version d'association n'aura pas d'expression de planification.

L'exemple suivant met à jour une association existante pour remplacer le nom par `TestHostnameAssociation2` . La nouvelle version d'association s'exécute toutes les heures et écrit la sortie des commandes dans le Amazon S3 compartiment spécifié.

Linux

```
aws ssm update-association \
  --association-id 8dfe3659-4309-493a-8755-01234EXAMPLE \
  --association-name TestHostnameAssociation2 \
  --parameters commands="echo Association" \
  --output-location S3Location='{OutputS3Region=us-east-1,OutputS3BucketName=DOC-
EXAMPLE-BUCKET,OutputS3KeyPrefix=logs}' \
  --schedule-expression "cron(0 */1 * * ? *)"
```

Windows

```
aws ssm update-association ^
  --association-id 8dfe3659-4309-493a-8755-01234EXAMPLE ^
  --association-name TestHostnameAssociation2 ^
  --parameters commands="echo Association" ^
  --output-location S3Location='{OutputS3Region=us-east-1,OutputS3BucketName=DOC-
EXAMPLE-BUCKET,OutputS3KeyPrefix=logs}' ^
  --schedule-expression "cron(0 */1 * * ? *)"
```

PowerShell

```
Update-SSMAssociation `
  -AssociationId b85ccafe-9f02-4812-9b81-01234EXAMPLE `
  -AssociationName TestHostnameAssociation2 `
  -Parameter @{"commands"="echo Association"} `
  -S3Location_OutputS3BucketName DOC-EXAMPLE-BUCKET `
  -S3Location_OutputS3KeyPrefix logs `
  -S3Location_OutputS3Region us-east-1 `
  -ScheduleExpression "cron(0 */1 * * ? *)"
```

L'exemple suivant met à jour une association existante pour remplacer le nom par `OneTimeAssociation` . La nouvelle version d'association ne s'exécute qu'une seule fois, à 15 h 55, heure UTC, le 7 juillet 2020, et écrit la sortie des commandes dans le Amazon S3 compartiment spécifié.

Linux

```
aws ssm update-association \
  --association-id 8dfe3659-4309-493a-8755-01234EXAMPLE \
  --association-name OneTimeAssociation \
  --parameters commands="echo Association" \
  --output-location S3Location='{OutputS3Region=us-east-1,OutputS3BucketName=DOC-
EXAMPLE-BUCKET,OutputS3KeyPrefix=logs}' \
  --schedule "at(2020-07-07T15:55:00)" \
  --apply-only-at-cron-interval
```

Windows

```
aws ssm update-association ^
  --association-id 8dfe3659-4309-493a-8755-01234EXAMPLE ^
  --association-name OneTimeAssociation ^
  --parameters commands="echo Association" ^
  --output-location S3Location='{OutputS3Region=us-east-1,OutputS3BucketName=DOC-
EXAMPLE-BUCKET,OutputS3KeyPrefix=logs}' ^
  --schedule "at(2020-07-07T15:55:00)" ^
  --apply-only-at-cron-interval
```

PowerShell

```
Update-SSMAssociation `
  -AssociationId b85ccafe-9f02-4812-9b81-01234EXAMPLE `
  -AssociationName OneTimeAssociation `
  -Parameter @{"commands"="echo Association"} `
  -S3Location_OutputS3BucketName DOC-EXAMPLE-BUCKET `
  -S3Location_OutputS3KeyPrefix logs `
  -S3Location_OutputS3Region us-east-1 `
  -ScheduleExpression "at(2020-07-07T15:55:00) `
  -ApplyOnlyAtCronInterval
```

3. Pour afficher la nouvelle version de l'association, exécutez la commande suivante.

   Linux

```
aws ssm describe-association \
  --association-id b85ccafe-9f02-4812-9b81-01234EXAMPLE
```

   Windows

```
aws ssm describe-association ^
  --association-id b85ccafe-9f02-4812-9b81-01234EXAMPLE
```

   PowerShell

```
Get-SSMAssociation `
  -AssociationId b85ccafe-9f02-4812-9b81-01234EXAMPLE | Select-Object *
```

   Le système renvoie des informations similaires à celles qui suivent.

   Linux

```
{
```

```
    "AssociationDescription": {
        "ScheduleExpression": "cron(0 */1 * * ? *)",
        "OutputLocation": {
            "S3Location": {
                "OutputS3KeyPrefix": "logs",
                "OutputS3BucketName": "DOC-EXAMPLE-BUCKET",
                "OutputS3Region": "us-east-1"
            }
        },
        "Name": "AWS-RunPowerShellScript",
        "Parameters": {
            "commands": [
                "echo Association"
            ]
        },
        "LastExecutionDate": 1559316400.338,
        "Overview": {
            "Status": "Success",
            "DetailedStatus": "Success",
            "AssociationStatusAggregatedCount": {}
        },
        "AssociationId": "b85ccafe-9f02-4812-9b81-01234EXAMPLE",
        "DocumentVersion": "$DEFAULT",
        "LastSuccessfulExecutionDate": 1559316400.338,
        "LastUpdateAssociationDate": 1559316389.753,
        "Date": 1559314038.532,
        "AssociationVersion": "2",
        "AssociationName": "TestHostnameAssociation2",
        "Targets": [
            {
                "Values": [
                    "Windows"
                ],
                "Key": "tag:Environment"
            }
        ]
    }
}
```

Windows

```
{
    "AssociationDescription": {
        "ScheduleExpression": "cron(0 */1 * * ? *)",
        "OutputLocation": {
            "S3Location": {
                "OutputS3KeyPrefix": "logs",
                "OutputS3BucketName": "DOC-EXAMPLE-BUCKET",
                "OutputS3Region": "us-east-1"
            }
        },
        "Name": "AWS-RunPowerShellScript",
        "Parameters": {
            "commands": [
                "echo Association"
            ]
        },
        "LastExecutionDate": 1559316400.338,
        "Overview": {
            "Status": "Success",
            "DetailedStatus": "Success",
            "AssociationStatusAggregatedCount": {}
        },
        "AssociationId": "b85ccafe-9f02-4812-9b81-01234EXAMPLE",
```

```
        "DocumentVersion": "$DEFAULT",
        "LastSuccessfulExecutionDate": 1559316400.338,
        "LastUpdateAssociationDate": 1559316389.753,
        "Date": 1559314038.532,
        "AssociationVersion": "2",
        "AssociationName": "TestHostnameAssociation2",
        "Targets": [
            {
                "Values": [
                    "Windows"
                ],
                "Key": "tag:Environment"
            }
        ]
    }
}
```

PowerShell

```
AssociationId                 : b85ccafe-9f02-4812-9b81-01234EXAMPLE
AssociationName               : TestHostnameAssociation2
AssociationVersion            : 2
AutomationTargetParameterName :
ComplianceSeverity            :
Date                          : 5/31/2019 2:47:18 PM
DocumentVersion               : $DEFAULT
InstanceId                    :
LastExecutionDate             : 5/31/2019 3:26:40 PM
LastSuccessfulExecutionDate   : 5/31/2019 3:26:40 PM
LastUpdateAssociationDate     : 5/31/2019 3:26:29 PM
MaxConcurrency                :
MaxErrors                     :
Name                          : AWS-RunPowerShellScript
OutputLocation                :
 Amazon.SimpleSystemsManagement.Model.InstanceAssociationOutputLocation
Overview                      :
 Amazon.SimpleSystemsManagement.Model.AssociationOverview
Parameters                    : {[commands,
 Amazon.Runtime.Internal.Util.AlwaysSendList`1[System.String]]}
ScheduleExpression            : cron(0 */1 * * ? *)
Status                        :
Targets                       : {tag:Environment}
```

Affichage des historiques des associations

Vous pouvez afficher toutes les exécutions correspondant à un ID d'association spécifique à l'aide de l'action DescribeAssociationExecutions d'API . Cette action vous permet de consulter rapidement l'état, le statut détaillé, les résultats, l'heure de la dernière exécution et d'autres informations relatives à une association State Manager. Cette action d'API inclut également des filtres pour vous aider à trouver rapidement les associations correspondant aux critères que vous spécifiez. Par exemple, vous pouvez spécifier une date et une heure précises, et utiliser un filtre GREATER_THAN pour n'afficher que les exécutions qui ont été traitées après la date et l'heure spécifiées.

Si, par exemple, une exécution d'association a échoué, vous pouvez explorer en détail une exécution spécifique à l'aide de l'action DescribeAssociationExecutionTargets d'API. Cette action affiche les ressources, telles que l'instance IDs , où l'association s'est exécutée et les différents statuts d'association. Vous pouvez ensuite constater rapidement quelle instance ou ressource n'est pas parvenue à exécuter une association. Vous pouvez utiliser l'ID de ressource pour afficher les détails de l'exécution d'une commande et identifier ainsi précisément l'étape de la commande qui a échoué.

Les exemples de cette section incluent également des informations sur la façon d'utiliser l'action StartAssociationsOnce d'API pour exécuter une association immédiatement et une seule fois. Vous pouvez utiliser cette action d'API lorsque vous examinez les exécutions d'associations qui ont échoué. Si vous constatez qu'une association a échoué, vous pouvez effectuer une modification sur la ressource, puis exécuter immédiatement cette association pour déterminer si la modification effectuée sur la ressource permet à l'association de s'exécuter correctement.

Affichage des historiques des associations (console)

Utilisez la procédure suivante pour afficher l'historique d'association correspondant à un ID d'association spécifique et afficher ensuite les détails de l'exécution pour une ou plusieurs ressources.

Pour afficher l'historique d'exécution correspondant à un ID d'association spécifique

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Sélectionnez State Manager.
3. Dans le champ ID d'association, choisissez l'association dont vous souhaitez afficher l'historique.
4. Choisissez le bouton Afficher les détails.
5. Choisissez l'onglet Historique d'exécution.
6. Choisissez une association dont vous souhaitez afficher les détails d'exécution au niveau des ressources. Par exemple, choisissez une association qui indique le statut Échec. Vous pouvez ensuite consulter les détails d'exécution pour les instances qui n'ont pas réussi à exécuter l'association.

   Note

   Utilisez les filtres de la zone de recherche pour rechercher l'exécution dont vous souhaitez afficher les détails.

7. Choisissez un ID d'exécution. La page Association execution targets (Cibles d'exécution de l'association) s'ouvre. Cette page affiche toutes les ressources qui ont exécuté l'association.
8. Choisissez un ID de ressource pour afficher les informations spécifiques relatives à cette ressource.

   Note

   Utilisez les filtres de la zone de recherche pour rechercher la ressource dont vous souhaitez afficher les détails.

9. Si vous effectuez une recherche sur une association dont l'exécution a échoué, vous pouvez utiliser le bouton Appliquer l'association maintenant pour exécuter une association immédiatement et une seule fois. Une fois que vous avez modifié la ressource sur laquelle l'exécution de l'association a échoué, choisissez le lien ID d'association dans le chemin de navigation.
10. Choisissez le bouton Appliquer l'association maintenant. Une fois l'exécution terminée, vérifiez que l'exécution de l'association a réussi.

Affichage des historiques d'associations (ligne de commande)

La procédure suivante décrit comment utiliser l'AWS CLI (sous Linux ou Windows) ou Outils AWS pour PowerShell pour afficher l'historique d'exécution d'un ID d'association spécifique. Ensuite, la procédure décrit comment afficher les détails d'exécution d'une ou de plusieurs ressources.

Pour afficher l'historique d'exécution correspondant à un ID d'association spécifique

1. Installez et configurez l'AWS CLI ou Outils AWS pour PowerShell si vous ne l'avez pas déjà fait.

Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

2. Exécutez la commande suivante pour afficher la liste des exécutions correspondant à un ID d'association spécifique.

   Linux

   ```
   aws ssm describe-association-executions \
     --association-id ID \
     --filters Key=CreatedTime,Value="2018-04-10T19:15:38.372Z",Type=GREATER_THAN
   ```

   > **Note**
   >
   > Cette commande inclut un filtre pour limiter les résultats aux exécutions qui se sont produites après une date et une heure spécifiques. Pour afficher toutes les exécutions correspondant à un ID d'association spécifique, supprimez le paramètre `--filters` et la valeur `Key=CreatedTime,Value="2018-04-10T19:15:38.372Z",Type=GREATER_THAN`.

   Windows

   ```
   aws ssm describe-association-executions ^
     --association-id ID ^
     --filters Key=CreatedTime,Value="2018-04-10T19:15:38.372Z",Type=GREATER_THAN
   ```

   > **Note**
   >
   > Cette commande inclut un filtre pour limiter les résultats aux exécutions qui se sont produites après une date et une heure spécifiques. Pour afficher toutes les exécutions correspondant à un ID d'association spécifique, supprimez le paramètre `--filters` et la valeur `Key=CreatedTime,Value="2018-04-10T19:15:38.372Z",Type=GREATER_THAN`.

   PowerShell

   ```
   Get-SSMAssociationExecution `
     -AssociationId ID `
     -Filter
    @{"Key"="CreatedTime";"Value"="2019-06-01T19:15:38.372Z";"Type"="GREATER_THAN"}
   ```

   > **Note**
   >
   > Cette commande inclut un filtre pour limiter les résultats aux exécutions qui se sont produites après une date et une heure spécifiques. Pour afficher toutes les exécutions correspondant à un ID d'association spécifique, supprimez le paramètre `-Filter` et la valeur `@{"Key"="CreatedTime";"Value"="2019-06-01T19:15:38.372Z";"Type"="GREATER_THAN"}`.

   Le système renvoie des informations similaires à celles qui suivent.

   Linux

   ```
   {
      "AssociationExecutions":[
         {
            "Status":"Success",
            "DetailedStatus":"Success",
   ```

```
            "AssociationId":"c336d2ab-09de-44ba-8f6a-6136cEXAMPLE",
            "ExecutionId":"76a5a04f-caf6-490c-b448-92c02EXAMPLE",
            "CreatedTime":1523986028.219,
            "AssociationVersion":"1"
        },
        {
            "Status":"Success",
            "DetailedStatus":"Success",
            "AssociationId":"c336d2ab-09de-44ba-8f6a-6136cEXAMPLE",
            "ExecutionId":"791b72e0-f0da-4021-8b35-f95dfEXAMPLE",
            "CreatedTime":1523984226.074,
            "AssociationVersion":"1"
        },
        {
            "Status":"Success",
            "DetailedStatus":"Success",
            "AssociationId":"c336d2ab-09de-44ba-8f6a-6136cEXAMPLE",
            "ExecutionId":"ecec60fa-6bb0-4d26-98c7-140308EXAMPLE",
            "CreatedTime":1523982404.013,
            "AssociationVersion":"1"
        }
    ]
}
```

Windows

```
{
    "AssociationExecutions":[
        {
            "Status":"Success",
            "DetailedStatus":"Success",
            "AssociationId":"c336d2ab-09de-44ba-8f6a-6136cEXAMPLE",
            "ExecutionId":"76a5a04f-caf6-490c-b448-92c02EXAMPLE",
            "CreatedTime":1523986028.219,
            "AssociationVersion":"1"
        },
        {
            "Status":"Success",
            "DetailedStatus":"Success",
            "AssociationId":"c336d2ab-09de-44ba-8f6a-6136cEXAMPLE",
            "ExecutionId":"791b72e0-f0da-4021-8b35-f95dfEXAMPLE",
            "CreatedTime":1523984226.074,
            "AssociationVersion":"1"
        },
        {
            "Status":"Success",
            "DetailedStatus":"Success",
            "AssociationId":"c336d2ab-09de-44ba-8f6a-6136cEXAMPLE",
            "ExecutionId":"ecec60fa-6bb0-4d26-98c7-140308EXAMPLE",
            "CreatedTime":1523982404.013,
            "AssociationVersion":"1"
        }
    ]
}
```

PowerShell

```
AssociationId         : c336d2ab-09de-44ba-8f6a-6136cEXAMPLE
AssociationVersion    : 1
CreatedTime           : 8/18/2019 2:00:50 AM
DetailedStatus        : Success
ExecutionId           : 76a5a04f-caf6-490c-b448-92c02EXAMPLE
LastExecutionDate     : 1/1/0001 12:00:00 AM
```

```
ResourceCountByStatus : {Success=1}
Status                : Success

AssociationId         : c336d2ab-09de-44ba-8f6a-6136cEXAMPLE
AssociationVersion    : 1
CreatedTime           : 8/11/2019 2:00:54 AM
DetailedStatus        : Success
ExecutionId           : 791b72e0-f0da-4021-8b35-f95dfEXAMPLE
LastExecutionDate     : 1/1/0001 12:00:00 AM
ResourceCountByStatus : {Success=1}
Status                : Success

AssociationId         : c336d2ab-09de-44ba-8f6a-6136cEXAMPLE
AssociationVersion    : 1
CreatedTime           : 8/4/2019 2:01:00 AM
DetailedStatus        : Success
ExecutionId           : ecec60fa-6bb0-4d26-98c7-140308EXAMPLE
LastExecutionDate     : 1/1/0001 12:00:00 AM
ResourceCountByStatus : {Success=1}
Status                : Success
```

Vous pouvez limiter les résultats en utilisant un ou plusieurs filtres. L'exemple suivant renvoie toutes les associations qui ont été exécutées avant une date et une heure spécifiques.

Linux

```
aws ssm describe-association-executions \
  --association-id ID \
  --filters Key=CreatedTime,Value="2018-04-10T19:15:38.372Z",Type=LESS_THAN
```

Windows

```
aws ssm describe-association-executions ^
  --association-id ID ^
  --filters Key=CreatedTime,Value="2018-04-10T19:15:38.372Z",Type=LESS_THAN
```

PowerShell

```
Get-SSMAssociationExecution `
  -AssociationId 14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE `
  -Filter
 @{"Key"="CreatedTime";"Value"="2019-06-01T19:15:38.372Z";"Type"="LESS_THAN"}
```

L'exemple suivant renvoie toutes les associations qui ont été exécutées correctement après une date et une heure spécifiques.

Linux

```
aws ssm describe-association-executions \
  --association-id ID \
  --filters Key=CreatedTime,Value="2018-04-10T19:15:38.372Z",Type=GREATER_THAN
 Key=Status,Value=Success,Type=EQUAL
```

Windows

```
aws ssm describe-association-executions ^
  --association-id ID ^
```

```
  --filters Key=CreatedTime,Value="2018-04-10T19:15:38.372Z",Type=GREATER_THAN
 Key=Status,Value=Success,Type=EQUAL
```

PowerShell

```
Get-SSMAssociationExecution `
  -AssociationId 14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE `
  -Filter @{
      "Key"="CreatedTime";
      "Value"="2019-06-01T19:15:38.372Z";
      "Type"="GREATER_THAN"
    },
    @{
      "Key"="Status";
      "Value"="Success";
      "Type"="EQUAL"
    }
```

3. Exécutez la commande suivante pour afficher toutes les cibles sur lesquelles l'exécution spécifique a eu lieu.

   Linux

```
aws ssm describe-association-execution-targets \
  --association-id ID \
  --execution-id ID
```

   Windows

```
aws ssm describe-association-execution-targets ^
  --association-id ID ^
  --execution-id ID
```

   PowerShell

```
Get-SSMAssociationExecutionTarget `
  -AssociationId 14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE `
  -ExecutionId 76a5a04f-caf6-490c-b448-92c02EXAMPLE
```

   Vous pouvez limiter les résultats en utilisant un ou plusieurs filtres. L'exemple suivant renvoie les informations relatives à toutes les cibles sur lesquelles l'exécution de l'association spécifique a échoué.

   Linux

```
aws ssm describe-association-execution-targets \
  --association-id ID \
  --execution-id ID \
  --filters Key=Status,Value="Failed"
```

   Windows

```
aws ssm describe-association-execution-targets ^
  --association-id ID ^
  --execution-id ID ^
  --filters Key=Status,Value="Failed"
```

PowerShell

```
Get-SSMAssociationExecutionTarget `
  -AssociationId 14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE `
  -ExecutionId 76a5a04f-caf6-490c-b448-92c02EXAMPLE `
  -Filter @{
      "Key"="Status";
      "Value"="Failed"
    }
```

L'exemple suivant renvoie les informations relatives à une instance gérée spécifique sur laquelle l'exécution d'une association a échoué.

Linux

```
aws ssm describe-association-execution-targets \
  --association-id ID \
  --execution-id ID \
  --filters Key=Status,Value=Failed Key=ResourceId,Value="i-02573cafcfEXAMPLE"
 Key=ResourceType,Value=ManagedInstance
```

Windows

```
aws ssm describe-association-execution-targets ^
  --association-id ID ^
  --execution-id ID ^
  --filters Key=Status,Value=Failed Key=ResourceId,Value="i-02573cafcfEXAMPLE"
 Key=ResourceType,Value=ManagedInstance
```

PowerShell

```
Get-SSMAssociationExecutionTarget `
  -AssociationId 14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE `
  -ExecutionId 76a5a04f-caf6-490c-b448-92c02EXAMPLE `
  -Filter @{
      "Key"="Status";
      "Value"="Success"
    },
    @{
      "Key"="ResourceId";
      "Value"="i-02573cafcfEXAMPLE"
    },
    @{
      "Key"="ResourceType";
      "Value"="ManagedInstance"
    }
```

4. Si vous effectuez une recherche sur une association dont l'exécution a échoué, vous pouvez utiliser l'action StartAssociationsOnce d'API pour exécuter une association immédiatement et une seule fois. Lorsque vous modifiez la ressource sur laquelle l'exécution de l'association a échoué, utilisez la commande suivante pour exécuter l'association immédiatement et une seule fois.

   Linux

```
aws ssm start-associations-once \
  --association-id ID
```

Windows

```
aws ssm start-associations-once ^
  --association-id ID
```

PowerShell

```
Start-SSMAssociationsOnce `
  -AssociationId ID
```

Procédures State Manager AWS Systems Manager

Les procédures suivantes démontrent comment créer et configurer les associations State Manager à l'aide de la console Systems Manager ou de l'AWS CLI. Elles peuvent aussi vous montrer comment exécuter automatiquement les tâches administratives à l'aide de State Manager.

Rubriques

- Création d'associations qui exécutent des fichiers MOF (p. 1019)
- Création d'associations exécutant des manuels stratégiques Ansible (p. 1030)
- Création d'associations qui exécutent des recettes de Chef (p. 1035)
- Mise à jour automatique d'Agent SSM (interface de ligne de commande) (p. 1043)
- Procédure : Mise à jour automatique des pilotes PV sur les instances EC2 pour Windows Server (console) (p. 1045)

Création d'associations qui exécutent des fichiers MOF

Vous pouvez exécuter des fichiers MOF (Managed Object Format) pour appliquer un état souhaité aux instances gérées de Windows Server avec State Manager en utilisant le document AWS-ApplyDSCMofs SSM. Le document AWS-ApplyDSCMofs a deux modes d'exécution. Avec le premier mode, vous pouvez configurer l'association pour analyser et indiquer si les instances gérées sont actuellement dans l'état souhaité défini dans les fichiers MOF spécifiés. Dans le second mode, vous pouvez exécuter les fichiers MOF et modifier la configuration de vos instances basées sur les ressources et leurs valeurs définies dans les fichiers MOF. Le document AWS-ApplyDSCMofs vous permet de télécharger et d'exécuter les fichiers de configuration MOF à partir de Amazon Simple Storage Service (Amazon S3), d'un partage local, ou à partir d'un site Web sécurisé avec un domaine HTTPS.

State Manager journalise et signale le statut de chaque exécution de fichier MOF au cours de chaque exécution d'association. State Manager signale également la sortie de chaque exécution de fichier MOF en tant qu'événement de conformité que vous pouvez afficher sur la page de conformité AWS Systems Manager.

L'exécution du fichier MOF repose sur la configuration PowerShell de l'état souhaité Windows (PowerShell DSC). PowerShellDSC est une plateforme déclarative utilisée pour la configuration, le déploiement et la gestion des systèmes Windows. PowerShellDSC permet aux administrateurs de décrire, dans des documents texte simples appelés configurations DSC, la façon dont ils souhaitent qu'un serveur soit configuré. Une configuration PowerShell DSC est un PowerShell script spécialisé qui indique ce qu'il faut faire, mais pas comment le faire. L'exécution de la configuration génère un fichier MOF. Le fichier MOF peuvent être appliqué à un ou plusieurs serveurs afin de réaliser la configuration souhaitée pour ces serveurs. Les ressources PowerShell DSC effectuent le travail réel d'application de la configuration. Pour plus d'informations, consultez Présentation de la configuration de l'état souhaité WindowsPowerShell.

Rubriques

- Utilisation d'Amazon S3 pour stocker les artefacts (p. 1020)
- Résolution des informations d'identification dans les fichiers MOF (p. 1020)
- Utilisation de jetons dans les fichiers MOF (p. 1021)
- Prerequisites (p. 1022)
- Création d'une association qui exécute des fichiers MOF (p. 1022)
- Troubleshooting (p. 1026)
- Affichage des détails de conformité des ressources DSC (p. 1028)

Utilisation d'Amazon S3 pour stocker les artefacts

Si vous utilisez Amazon S3 pour stocker PowerShell des modules, des fichiers MOF, des rapports de conformité ou des rapports d'état, le IAM rôle utilisé par Agent SSM doit détenir `GetObject` les autorisations `ListBucket` et sur le compartiment. Si vous ne fournissez pas ces autorisations, le système renvoie une erreur Accès refusé. Notez également les informations importantes suivantes concernant le stockage d'artefacts dans Amazon S3.

- Si le compartiment se trouve dans un autre compte AWS, vous devez créer une stratégie de ressource de compartiment qui accorde au compte (ou au rôle IAM) les autorisations `GetObject` et `ListBucket`.
- Si vous souhaitez utiliser des ressources DSC personnalisées, vous pouvez télécharger ces ressources à partir d'un compartiment Amazon S3. Vous pouvez également les installer automatiquement à partir de la PowerShell galerie.
- Si vous utilisez Amazon S3 comme module source, vous devez charger le module en tant que fichier Zip dans le format sensible à la casse suivant : *`ModuleName_ModuleVersion`*.zip. Exemple : MyModule: _1.0.0.zip.
- Tous les fichiers doivent se trouver dans le compartiment racine. Les structures de dossier ne sont pas prises en charge.

Résolution des informations d'identification dans les fichiers MOF

Les informations d'identification sont résolues en utilisant AWS Secrets Manager ou AWS Systems Manager Parameter Store (p. 218). Cela vous permet de configurer la rotation automatique des informations d'identification. Cela permet également à DSC de propager automatiquement les informations d'identification à vos serveurs sans redéployer MOFs .

Pour utiliser un Secrets Manager secret dans une configuration, créez un PSCredential objet où le nom d'utilisateur est le SecretId ou le secret contenant les informations d'identification.SecretARN Vous pouvez spécifier n'importe quelle valeur pour le mot de passe. La valeur est ignorée. Voici un exemple :

```
Configuration MyConfig
{
   $ss = ConvertTo-SecureString -String 'a_string' -AsPlaintext -Force
   $credential = New-Object PSCredential('a_secret_or_ARN', $ss)

    Node localhost
    {
       File file_name
       {
           DestinationPath = 'C:\MyFile.txt'
           SourcePath = '\\FileServer\Share\MyFile.txt'
           Credential = $credential
       }
    }
}
```

Vous devez ensuite compiler votre MOF en utilisant le paramètre PsAllowPlaintextPassword dans les données de configuration. Cette opération ne pose pas de problème, car les informations d'identification contiennent uniquement une étiquette.

Dans Secrets Manager , assurez-vous que l'instance a GetSecretValue accès dans une stratégie gérée IAM et, le cas échéant, dans la stratégie de ressources du secret, le cas échéant. Pour fonctionner avec DSC, le secret doit être au format suivant :

```
{ 'Username': 'a_name', 'Password': 'a_password' }
```

Le secret peut avoir d'autres propriétés (par exemple, des propriétés utilisées pour la rotation), mais il doit comporter au moins le nom d'utilisateur et le mot de passe.

Nous vous recommandons d'utiliser une méthode de rotation multi-utilisateur, avec laquelle vous avez deux noms d'utilisateur et deux mots de passe différents, la fonction AWS Lambda effectuant une rotation entre eux. Cette méthode vous permet d'avoir plusieurs comptes actifs tout en éliminant le risque de bloquer un utilisateur pendant la rotation.

Utilisation de jetons dans les fichiers MOF

Les jetons vous permettent de modifier des valeurs de propriété de ressource après que le fichier MOF a été compilé. Cela vous permet de réutiliser des fichiers MOF courants sur plusieurs serveurs qui nécessitent des configurations similaires.

La substitution de jeton fonctionne uniquement pour les propriétés de ressource de type `String` . Toutefois, si votre ressource a une propriété d'instance CIM imbriquée, elle résoudra également les jetons à partir des `String` propriétés de cette instance CIM. Vous ne pouvez pas utiliser la substitution de jeton pour des nombres ou des tableaux.

Prenons l'exemple d'un scénario dans lequel vous utilisez la xComputerManagement ressource et souhaitez renommer l'ordinateur à l'aide de DSC. Normalement, vous avez besoin d'un fichier MOF dédié pour cette machine. Cependant, avec la prise en charge des jetons, vous pouvez créer un seul fichier MOF et l'appliquer à toutes vos instances. Dans la propriété `ComputerName`, au lieu de coder en dur le nom d'ordinateur dans le fichier MOF, vous pouvez utiliser un jeton de type balise (Tag) d'instance. La valeur est résolue lors de l'analyse du fichier MOF. Exemple :

```
Configuration MyConfig
{
    xComputer Computer
    {
        ComputerName = '{tag:ComputerName}'
    }
}
```

Vous définissez ensuite une balise sur l'instance gérée dans la console AWS Systems Manager, ou une balise Amazon EC2 dans la console EC2. Lorsque vous exécutez le document, le script substitue le jeton {tag:ComputerName} à la valeur de la balise d'instance.

Vous pouvez également combiner plusieurs balises dans une seule propriété, par exemple :

```
Configuration MyConfig
{
    File MyFile
    {
        DestinationPath = '{env:TMP}\{tag:ComputerName}'
        Type = 'Directory'
    }
}
```

Vous pouvez utiliser 5 types de jetons différents :

- tag : Balises Amazon EC2 ou d'instance gérée
- tagb64 : Identique à la balise, mais le système utilise base64 pour décoder la valeur. Cela vous permet d'utiliser des caractères spéciaux dans des valeurs de balise.
- env : Résout les variables d'environnement.
- ssm: Valeurs Systems Manager Parameter Store. Seuls les types String et Secure String sont pris en charge.
- balises : Identique à la balise, mais si la balise n'est pas définie sur l'instance, le système tente de résoudre la valeur à partir d'un paramètre SSM avec le même nom. Cela s'avère utile dans des situations où vous souhaitez disposer d'une « valeur globale par défaut », mais pouvoir la remplacer sur une seule instance (par exemple, pour des déploiements « one-box »).

Voici un exemple Parameter Store qui utilise le type de jeton `ssm`.

```
File MyFile
{
    DestinationPath = "C:\ProgramData\ConnectionData.txt"
    Content = "{ssm:%servicePath%/ConnectionData}"
}
```

Les jetons jouent un rôle important pour la réduction du code redondant en rendant les fichiers MOF génériques et réutilisables. Si vous pouvez éviter les fichiers MOF spécifiques à un serveur, vous n'avez pas besoin d'un service de génération de fichier MOF. Un service de génération de fichier MOF augmente les coûts, le temps de mise en service et le risque d'écart de configuration entre les instances regroupées en raison de l'installation de versions de module différentes sur le serveur de génération lors de leur compilation.MOFs

Prerequisites

Avant de créer une association qui exécute des fichiers MOF, vérifiez que vos instances gérées ont les prérequis suivants installés :

- PowerShellWindows version 5.0 ou ultérieure. Pour plus d'informations, consultez Configuration PowerShell système requise pour Windows sur Microsoft.com.
- Outils AWS pour Windows PowerShell version 3.3.261.0 ou ultérieure.
- Agent SSM version 2.2 ou une version ultérieure.

Création d'une association qui exécute des fichiers MOF

Pour créer une association qui exécute des fichiers MOF

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez State Manager, puis Créer une association.
3. Dans le champ Nom, spécifiez un nom. Cette action est facultative, mais recommandée. Un nom peut vous aider à comprendre quel était l'objectif de l'association quand vous avez créée celle-ci. Les espaces ne sont pas autorisés dans le nom.
4. Dans la liste Document, choisissez AWS-ApplyDSCMofs.
5. Dans la section Parameters (Paramètres), spécifiez vos choix pour les paramètres d'entrée obligatoires et facultatifs.

   a. Mofs à appliquer : Spécifiez un ou plusieurs fichiers MOF à exécuter lorsque cette association s'exécute. Utilisez des virgules pour séparer les fichiers MOF dans une liste. Vous pouvez spécifier les options suivantes pour localiser un fichier MOF.

- Un nom de compartiment Amazon S3. Les noms de compartiment doivent utiliser des lettres minuscules. Spécifiez cette information à l'aide du format suivant :

```
s3:doc-example-bucket:MOF_file_name.mof
```

Si vous souhaitez spécifier une région AWS, utilisez le format suivant :

```
s3:bucket_Region:doc-example-bucket:MOF_file_name.mof
```

- Un site web sécurisé. Spécifiez cette information à l'aide du format suivant :

```
https://domain_name/MOF_file_name.mof
```

Voici un exemple :

```
https://www.example.com/TestMOF.mof
```

- Un système de fichiers sur un partage local. Spécifiez cette information à l'aide du format suivant :

```
\server_name\shared_folder_name\MOF_file_name.mof
```

Voici un exemple :

```
\StateManagerAssociationsBox\MOFs_folder\MyMof.mof
```

b. Chemin d'accès au service : (Facultatif) Un chemin d'accès au service est un préfixe de Amazon S3 compartiment dans lequel vous souhaitez écrire des rapports et des informations de statut. Ou, un chemin d'accès au service est un chemin pour des balises basées sur des paramètres Parameter Store. Lors de la résolution des balises basées sur les paramètres, le système utilise {ssm:%servicePath%/*parameter_name*} pour injecter la servicePath valeur dans le nom du paramètre. Par exemple, si le chemin d'accès au service est « Serveurs Web/Production », le système résout le paramètre comme suit : WebServers/Production/*parameter_name*. Cela est utile lorsque vous exécutez plusieurs environnements dans le même compte.

c. Nom du compartiment du rapport : (Facultatif) Entrez le nom d'un Amazon S3 compartiment dans lequel vous souhaitez écrire les données de conformité. Les rapports sont enregistrés dans ce compartiment au format JSON.

> **Note**
>
> Vous pouvez préfixer le nom de compartiment avec une région dans laquelle se trouve le compartiment. Voici un exemple : us-west-2:MyMOFBucket. Si vous utilisez un proxy pour les points de terminaison Amazon S3 dans une région spécifique qui n'inclut pas us-east-1, vous devez préfixer le nom de compartiment avec une région. Si le nom du compartiment n'est pas préfixé, la région du compartiment sera automatiquement découverte à l'aide du point de terminaison us-east-1.

d. Mode d'opération Mof : Choisissez State Manager le comportement lors de l'exécution de l'association AWS-ApplyDSCMofs :

- S'appliquer : Configurations d'instance correctes qui ne sont pas conformes.
- ReportOnly: Ne corrigez pas les configurations d'instance, mais consignez plutôt toutes les données de conformité et signalez les instances qui ne sont pas conformes.

e. Nom du compartiment d'état : (Facultatif) Entrez le nom d'un Amazon S3 compartiment dans lequel vous souhaitez écrire les informations relatives au statut d'exécution du fichier MOF. Ces

rapports de statut sont des résumés de singleton de la dernière exécution de conformité d'une instance. Cela signifie que le rapport est remplacé la fois suivante où l'association exécute des fichiers MOF.

> **Note**
>
> Vous pouvez préfixer le nom de compartiment avec une région dans laquelle se trouve le compartiment. Voici un exemple: `us-west-2:doc-example-bucket`. Si vous utilisez un proxy pour les Amazon S3 points de terminaison d'une région spécifique qui n'inclut pas us-east-1, vous devez préfixer le nom du compartiment par une région. Si le nom du compartiment n'est pas préfixé, la région du compartiment sera automatiquement découverte à l'aide du point de terminaison us-east-1.

f. Nom du compartiment source du module : (Facultatif) Entrez le nom d'un Amazon S3 compartiment qui contient les fichiers de PowerShell module. Si vous spécifiez None (Aucun), vous devez choisir True (Vrai) pour l'option suivante, Allow PS Gallery Module Source (Autoriser la galerie PS comme source de module).

> **Note**
>
> Vous pouvez préfixer le nom de compartiment avec une région dans laquelle se trouve le compartiment. Voici un exemple: `us-west-2:doc-example-bucket`. Si vous utilisez un proxy pour les Amazon S3 points de terminaison d'une région spécifique qui n'inclut pas us-east-1, vous devez préfixer le nom du compartiment avec une région. Si le nom du compartiment n'est pas préfixé, la région du compartiment sera automatiquement découverte à l'aide du point de terminaison us-east-1.

g. Autoriser la galerie PS comme source de module : (Facultatif) Choisissez True pour télécharger PowerShell les modules depuis https://www.powershellgallery.com/. Si vous choisissez False, vous devez spécifier une source pour l'option précédente, ModuleSourceBucketName .

h. URI du proxy : (Facultatif) Utilisez cette option pour télécharger les fichiers MOF à partir d'un serveur proxy.

i. Comportement du redémarrage : (Facultatif) Spécifiez l'un des comportements de redémarrage suivants si l'exécution de votre fichier MOF nécessite un redémarrage :

   - AfterMof: Il redémarre l'instance une fois toutes les exécutions de fichier MOF terminées. Même si plusieurs exécutions de fichier MOF demandent un redémarrage, le système attend que toutes les exécutions de fichier MOF soient terminées pour redémarrer.
   - Immédiatement : Redémarre l'instance chaque fois qu'une exécution de fichier MOF le demande. Lors de l'exécution de plusieurs fichiers MOF demandant un redémarrage, l'instance sera redémarrée plusieurs fois.
   - Never (Jamais) : Les instances ne sont pas redémarrées, même si l'exécution MOF demande explicitement un redémarrage.

j. Utiliser le nom de l'ordinateur pour les rapports : (Facultatif) Activez cette option pour utiliser le nom de l'ordinateur lors de la génération des informations de conformité. La valeur par défaut est false (faux), qui signifie que le système utilise l'ID d'instance lors de la création des rapports de conformité.

k. Activer la journalisation détaillée : (Facultatif) Nous vous recommandons d'activer la journalisation détaillée lorsque vous déployez des fichiers MOF pour la première fois.

> **Important**
>
> Lorsqu'elle est activée, la journalisation détaillée écrit plus de données dans votre compartiment Amazon S3 que la journalisation d'exécution d'association standard. Cela peut entraîner un ralentissement des performances et éventuellement des frais de stockage plus élevés pour Amazon S3. Pour éviter les problèmes de taille de stockage, nous vous recommandons d'activer des stratégies de cycle de vie sur le compartiment Amazon S3. Pour plus d'informations, consultez Comment créer la stratégie de cycle de

vie d'un compartiment S3 ? dans le Amazon Simple Storage Service Guide de l'utilisateur de la console.

l. Activer la journalisation de débogage : (Facultatif) Nous vous recommandons d'activer la journalisation de débogage si vous avez besoin de résoudre les échecs de fichier MOF. Nous vous recommandons également de désactiver cette option pour une utilisation normale.

> **Important**
>
> Lorsqu'elle est activée, la journalisation de débogage écrit plus de données dans votre compartiment Amazon S3 que la journalisation d'exécution d'association standard. Cela peut entraîner un ralentissement des performances et éventuellement des frais de stockage plus élevés pour Amazon S3. Pour éviter les problèmes de taille de stockage, nous vous recommandons d'activer des stratégies de cycle de vie sur le compartiment Amazon S3. Pour plus d'informations, consultez Comment créer la stratégie de cycle de vie d'un compartiment S3 ? dans le Amazon Simple Storage Service Guide de l'utilisateur de la console.

m. Type de conformité : (Facultatif) Spécifiez le type de conformité à utiliser lors de la génération de rapports sur les informations de conformité. Le type de conformité par défaut est Custom:DSC. Si vous créez plusieurs associations qui exécutent des fichiers MOF, assurez-vous de spécifier un type de conformité différent pour chaque association. Sinon, chaque association supplémentaire qui utilise Custom:DSC écrasera les données de conformité existantes.

n. Script avant redémarrage : (Facultatif) Spécifiez un script à exécuter si la configuration a indiqué qu'un redémarrage était nécessaire. Le script s'exécute avant le redémarrage. Le script doit tenir sur une seule ligne. Si vous devez ajouter des lignes supplémentaires, séparez-les avec des points-virgules.

6. Dans la section Targets (Cibles), choisissez Specifying tags (Spécification de balises) ou Manually Selecting Instance (Sélection manuelle des instances). Si vous choisissez de cibler des ressources à l'aide de balises, entrez une clé de balise et une valeur de balise dans les champs fournis. Pour plus d'informations sur l'utilisation des cibles, consultez À propos des cibles et des contrôles du débit dans les associations State Manager (p. 990).
7. Dans la section Specify schedule (Spécifier le programme), choisissez On Schedule (Selon le calendrier) ou No schedule (Pas de calendrier). Si vous choisissez On Schedule (Selon le calendrier), utilisez les boutons fournis pour créer un calendrier de type cron ou rate pour l'association.
8. Dans la section Advanced options (Options avancées) :
   - Dans Compliance severity (Gravité de conformité), choisissez un niveau de gravité pour l'association. Les rapports de conformité indiqueront si l'état de l'association est conforme ou non conforme, ainsi que le niveau de gravité que vous spécifiez ici. Pour plus d'informations, consultez la section A propos de la conformité des associations State Manager (p. 783).
9. Dans la section Rate control (Contrôle de taux), configurez des options pour l'exécution d'associations State Manager dans un parc d'instances gérées. Pour de plus amples informations sur ces options, veuillez consulter À propos des cibles et des contrôles du débit dans les associations State Manager (p. 990).

   Dans la section Simultanéité, choisissez une option :

   - Choisissez targets (cibles) pour entrer un nombre absolu de cibles pouvant exécuter l'association simultanément.
   - Choisissez percentage (pourcentage) pour saisir un pourcentage de l'ensemble de cibles pouvant exécuter l'association simultanément.

   Dans la section Error threshold (Seuil d'erreurs), choisissez une option :

   - Choisissez errors (erreurs) pour saisir un nombre absolu d'erreurs autorisées avant que State Manager cesse de d'exécuter des associations sur des cibles supplémentaires.

- Choisissez percentage (pourcentage) pour saisir un pourcentage d'erreurs autorisées avant que State Manager cesse de d'exécuter des associations sur des cibles supplémentaires.

10. (Facultatif) Dans Output options (Options de sortie), pour enregistrer la sortie de la commande dans un fichier, sélectionnez Enable writing to an S3 bucket (Autoriser l'écriture dans un compartiment S3) Entrez les noms de compartiment et de préfixe (dossier) dans les zones.

    Note

    Les autorisations S3 qui accordent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour de plus amples informations, veuillez consulter Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). En outre, si le compartiment S3 spécifié se trouve dans un compte AWS différent, assurez-vous que le profil d'instance associé à l'instance dispose des autorisations nécessaires pour écrire dans ce compartiment.

11. Choisissez Create Association (Créer une association).

State Manager crée et exécute immédiatement l'association sur les instances ou les cibles spécifiées. Après l'exécution initiale, l'association s'exécute à des intervalles selon le programme que vous avez défini et les règles suivantes :

- Les associations sont exécutées uniquement sur les instances en ligne lorsque l'intervalle commence. Les instances hors connexion sont ignorées.
- State Manager tente d'exécuter l'association sur toutes les instances configurées au cours d'un intervalle.
- Si une association n'est pas exécutée au cours d'un intervalle (par exemple, parce qu'une valeur de simultanéité a limité par le nombre d'instances pouvant traiter l'association simultanément), State Manager tente d'exécuter l'association lors du prochain intervalle.
- State Manager enregistre un historique pour tous les intervalles ignorés. Vous pouvez afficher l'historique dans l'onglet Execution History (Historique d'exécution).

Note

AWS ApplyDSCMofs est un document de commande Systems Manager. Cela signifie que vous pouvez également exécuter ce document en utilisant Fonctionnalité Exécuter la commande. Pour plus d'informations, consultez la section Exécuter des commandes avec Systems Manager Fonctionnalité Exécuter la commande (p. 946).

Troubleshooting

Cette section comprend des informations qui vous aideront à résoudre les problèmes liés à la création d'associations qui exécutent des fichiers MOF.

Activation de la journalisation améliorée

Comme première étape du dépannage, activez la journalisation améliorée. Plus précisément, procédez comme suit :

- Vérifiez que l'association est configurée pour écrire la sortie de commande dans Amazon S3 ou Amazon CloudWatch Logs.
- Définissez le paramètre Enable Verbose Logging (Activer la journalisation détaillée) sur True.
- Définissez le paramètre Enable Debug Logging (Activer la journalisation de débogage) sur True.

Avec la journalisation détaillée et de débogage activée, le fichier de sortie Stdout comprend plus de détails sur l'exécution du script. Ce fichier de sortie peut vous aider à identifier l'endroit où le script a échoué. Le fichier de sortie Stderr contient les erreurs qui se sont produites au cours de l'exécution du script.

Problèmes courants

Cette section inclut des informations sur les problèmes courants qui peuvent se produire lorsque vous créez des associations qui exécutent des fichiers MOF. Il comprend également les étapes permettant de résoudre ces problèmes.

Mon fichier MOF n'a pas été appliqué

Si State Manager n'a pas pu appliquer l'association à vos instances, commencez par vérifier fichier de sortie Stderr. Ce fichier peut vous aider à comprendre la cause première du problème. Vérifiez également que les points suivants :

- L'instance comporte les autorisations d'accès requises à tous les compartiments Amazon S3 liés aux fichiers MOF. En particulier :
  - s3:Autorisations GetObject : Il est obligatoire pour les fichiers MOF dans des Amazon S3 compartiments privés, ainsi que pour les modules personnalisés dans des Amazon S3 compartiments.
  - Autorisation s3:PutObject : Ceci est nécessaire pour écrire des rapports de conformité et l'état de conformité dans les Amazon S3 compartiments .
- Si vous utilisez des balises, assurez-vous que l'instance dispose des stratégies IAM requises. L'utilisation de balises nécessite que le rôle IAM d'instance dispose d'une stratégie autorisant les actions `ec2:DescribeInstances` et `ssm:ListTagsForResource`.
- Assurez-vous que l'instance dispose des balises attendues ou des paramètres SSM attribués.
- Assurez-vous que les balises ou les paramètres SSM sont correctement orthographiés.
- Essayez d'appliquer le fichier MOF localement sur l'instance pour vous assurer que le problème n'est pas lié au fichier MOF lui-même.

Mon fichier MOF semble avoir échoué, mais l'exécution Systems Manager a réussi

Si le document AWS-ApplyDSCMofs s'est exécuté avec succès, le statut d'exécution Systems Manager affiche Success (Réussite). Ce statut ne reflète pas le statut de conformité de votre instance par rapport aux exigences de configuration figurant dans le fichier MOF. Pour voir le statut de conformité de vos instances, consultez les rapports de conformité. Vous pouvez afficher un rapport JSON dans le compartiment des rapports Amazon S3. Cela s'applique aux exécutions Fonctionnalité Exécuter la commande et State Manager. En outre, pour State Manager, vous pouvez afficher les détails de conformité sur la page de conformité Systems Manager.

États Stderr : Échec de la résolution de nom lors de la tentative d'accès au service

Cette erreur indique que le script ne peut pas atteindre un service distant. Il est vraisemblable que le script ne peut pas atteindre Amazon S3. Ce problème se produit le plus souvent lorsque le script tente d'écrire des rapports de conformité ou un statut de conformité dans le compartiment Amazon S3 fourni dans les paramètres du document. En général, cette erreur se produit lorsqu'un environnement informatique utilise un pare-feu ou un proxy transparent qui inclut une liste d'autorisation. Pour résoudre ce problème :

- Utilisez la syntaxe de compartiment spécifique à la région pour tous les paramètres de compartiment Amazon S3. Par exemple, le paramètre Mofs To Apply (Fichiers MOF à appliquer) doit être formaté comme suit :

  s3 :*bucket-region*:*bucket-name*:*mof-file-name*.

  Voici un exemple : `s3:us-west-2:doc-example-bucket:my-mof.mof`

  Les noms des compartiments de rapport, de statut et de source de module doivent être formatés comme suit :

  bucket-region:*bucket-name*. Voici un exemple :`us-west-1:doc-example-bucket`

- Si la syntaxe spécifique à la région ne permet pas de résoudre le problème, assurez-vous que la ou les instances ciblées peuvent accéder à Amazon S3 dans la région souhaitée. Pour vérifier ceci :
  1. Recherchez le nom du point de terminaison pour Amazon S3 dans la région Amazon S3 appropriée. Pour plus d'informations, consultez Points de terminaison du service Amazon S3 dans le Référence générale d'Amazon Web Services.
  2. Connectez-vous à l'instance cible et exécutez la commande ping suivante :

     ```
     ping s3.s3-region.amazonaws.com
     ```

     Si le ping a échoué, cela signifie qu'Amazon S3 est arrêté, qu'un pare-feu/proxy transparent bloque l'accès à la région Amazon S3, ou que l'instance ne peut pas accéder à Internet.

Affichage des détails de conformité des ressources DSC

Systems Manager capture les informations de conformité sur les défaillances de ressources DSC dans le Status Bucket (Compartiment de statut) Amazon Simple Storage Service (Amazon S3) que vous avez spécifié lorsque vous avez exécuté le document ApplyDSCMofs. La recherche d'informations sur les défaillances de ressources DSC dans un compartiment Amazon S3 peut prendre du temps. Au lieu de cela, vous pouvez consulter rapidement ces informations sur la page Systems Manager Conformité.

La section Récapitulatif des ressources de conformité affiche le nombre de ressources qui ont échoué. Dans l'exemple suivant, la ComplianceType est Custom:DSC et une ressource n'est pas conforme.

> **Note**
>
> Custom:DSC est la ComplianceType valeur par défaut dans le document AWS-ApplyDSCMofs. Cette valeur est personnalisable.

La section Présentation détaillée des ressources affiche des informations sur la ressource AWS avec la ressource DSC non conforme. Cette section inclut également le nom MOF, les étapes d'exécution du script et (le cas échéant) un lien View output (Afficher la sortie) pour consulter des informations de statut détaillées.

Le lien View output (Afficher la sortie) affiche les derniers 4000 caractères du statut détaillé. Systems Manager commence par l'exception comme premier élément, puis analyse les messages détaillés et en ajoute le plus possible jusqu'à la limite de 4000 caractères. Ce processus affiche les messages de journal qui ont été générés avant que l'exception soit déclenchée, qui sont les plus pertinents pour la résolution.

Pour de plus amples informations sur la façon d'afficher les informations de conformité, veuillez consulter Conformité de la configuration AWS Systems Manager (p. 778).

Situations qui affectent les rapports de conformité

Si l'association State Manager échoue, aucune donnée de conformité n'est présentée. Plus spécifiquement, si un MOF ne peut pas être traité, Systems Manager ne signale aucun élément de conformité, car les associations échouent. Par exemple, si Systems Manager tente de télécharger un MOF à partir d'un compartiment Amazon S3 auquel l'instance n'est pas autorisé à accéder, l'association échoue et aucune donnée de conformité n'est présentée.

Si une ressource dans un deuxième MOF échoue, Systems Manager ne génère pas des donnée de conformité. Par exemple, si un MOF tente de créer un fichier sur un lecteur qui n'existe pas, Systems Manager signale la conformité, car le document AWS-ApplyDSCMofs peut être traité complètement, ce qui signifie que l'association s'exécute avec succès.

Création d'associations exécutant des manuels stratégiques Ansible

Vous pouvez créer des associations State Manager qui exécutent des manuels stratégiques Ansible à l'aide du document `AWS-ApplyAnsiblePlaybooks`. Ce document offre les avantages suivants pour l'exécution de manuels stratégiques :

- Prise en charge de l'exécution de manuels stratégiques complexes
- Prise en charge du téléchargement de manuels stratégiques depuis GitHub et Amazon Simple Storage Service (Amazon S3)
- Prise en charge de la structure de manuel stratégique compressé
- Journalisation améliorée
- Possibilité de spécifier le manuel stratégique à exécuter lorsque les manuels stratégiques sont regroupés

> **Note**
>
> Systems Managerinclut deux documents SSM qui vous permettent de créer State Manager des associations exécutant des manuels stratégiques Ansible : `AWS-RunAnsiblePlaybook`et `AWS-ApplyAnsiblePlaybooks` . Le `AWS-RunAnsiblePlaybook` document est obsolète. Il reste disponible dans Systems Manager à des fins de conservation. Nous vous recommandons d'utiliser le document `AWS-ApplyAnsiblePlaybooks` en raison des améliorations décrites ici.

Prise en charge de l'exécution de manuels stratégiques complexes

Le document `AWS-ApplyAnsiblePlaybooks` prend en charge les manuels stratégiques complexes et groupés, car il copie toute la structure de fichiers dans un répertoire local avant d'exécuter le manuel stratégique principal spécifié. Vous pouvez fournir des manuels stratégiques source dans des fichiers Zip ou dans une structure de répertoires. Le fichier Zip ou le répertoire peut être stocké dans GitHub ou Amazon S3 .

Prise en charge du téléchargement de manuels stratégiques à partir de GitHub

Le document `AWS-ApplyAnsiblePlaybooks` utilise le plug-in `aws:downloadContent` pour télécharger les fichiers du manuel stratégique. Les fichiers peuvent être stockés dans GitHub un seul fichier ou sous la forme d'un ensemble combiné de fichiers de manuel stratégique. Pour télécharger du contenu depuis GitHub , vous devez spécifier les informations relatives à votre GitHub référentiel au format JSON. Voici un exemple :

```
{
   "owner":"TestUser",
   "repository":"GitHubTest",
   "path":"scripts/python/test-script",
   "getOptions":"branch:master",
```

```
   "tokenInfo":"{{ssm-secure:secure-string-token}}"
}
```

Prise en charge du téléchargement de manuels stratégiques à partir de Amazon S3

Vous pouvez également stocker et télécharger les manuels stratégiques Ansible dans Amazon S3 sous la forme d'un fichier .zip unique ou d'une structure de répertoires. Pour télécharger du contenu depuis Amazon S3, vous devez spécifier le chemin d'accès au fichier. Voici deux exemples :

Exemple 1 : Téléchargement d'un fichier de manuel stratégique spécifique

```
{
   "path":"https://s3.amazonaws.com/doc-example-bucket/playbook.yml"
}
```

Exemple 2 : Téléchargement du contenu d'un répertoire

```
{
   "path":"https://s3.amazonaws.com/doc-example-bucket/ansible/webservers/"
}
```

> **Important**
>
> Si vous spécifiez Amazon S3, le profil d'instance AWS Identity and Access Management (IAM) sur vos instances gérées doit être configuré avec la stratégie `AmazonS3ReadOnlyAccess`. Pour plus d'informations, consultez la section Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34).

Prise en charge de la structure de manuel stratégique compressé

Le document `AWS-ApplyAnsiblePlaybooks` vous permet d'exécuter des fichiers .zip compressés dans le bundle téléchargé. Le document vérifie si les fichiers téléchargés contiennent un fichier compressé au format .zip. Si un fichier .zip est trouvé, le document décompresse automatiquement le fichier, puis exécute l'automatisation Ansible spécifiée.

Journalisation améliorée

Le document `AWS-ApplyAnsiblePlaybooks` inclut un paramètre facultatif pour spécifier différents niveaux de journalisation. Spécifiez -v pour une journalisation avec un niveau de détail faible, -vv ou -vvv pour une journalisation avec un niveau de détail moyen et -vvvv pour une journalisation avec u niveau de débogage. Ces options mappent directement sur les options de niveau de détail Ansible.

Possibilité de spécifier le manuel stratégique à exécuter lorsque les manuels stratégiques sont regroupés

Le document `AWS-ApplyAnsiblePlaybooks` inclut un paramètre obligatoire pour spécifier le manuel stratégique à exécuter lorsque plusieurs manuels sont regroupés. Cette option offre une flexibilité pour exécuter des manuels stratégiques afin de prendre en charge différents cas d'utilisation.

Dépendances installées

Si vous spécifiez True pour le InstallDependencies paramètre, Systems Manager vérifie que les dépendances suivantes sont installées sur vos instances. Si une ou plusieurs de ces dépendances sont introuvables, Systems Manager les installe automatiquement.

- Ubuntu Server/ Debian Server : Apt-get (gestion de package), Python 3, Ansible, Unzip
- Amazon Linux : Ansible
- RHEL : Python 3, Ansible, Décompresser

Créer une association exécutant des manuels stratégiques Ansible (console)

La procédure suivante décrit comment utiliser la console Systems Manager pour créer une association State Manager qui exécute les manuels stratégiques Ansible à l'aide du document `AWS-ApplyAnsiblePlaybooks`.

Pour créer une association qui exécute les manuels stratégiques Ansible (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez State Manager, puis Créer une association.
3. Pour Name (Nom), spécifiez un nom qui vous aide à mémoriser l'objectif de l'association.
4. Dans la liste Document choisissez AWS-ApplyAnsiblePlaybooks.
5. Dans la section Parameters (Paramètres), pour Source Type (Type de source), choisissez GitHub ou S3.

   GitHub

   Si vous choisissez GitHub , entrez les informations du référentiel au format suivant :

   ```
   {
      "owner":"user_name",
      "repository":"name",
      "path":"path_to_directory_or_playbook_to_download",
      "getOptions":"branch:branch_name",
      "tokenInfo":"{{(Optional)_token_information}}"
   }
   ```

   S3

   Si vous choisissez S3, entrez les informations de chemin au format suivant :

   ```
   {
      "path":"https://s3.amazonaws.com/path_to_directory_or_playbook_to_download"
   }
   ```
6. Pour Install Dependencies (Installer des dépendances), choisissez une option.
7. (Facultatif) Pour Playbook File (Fichier de manuel stratégique), entrez un nom de fichier. Si le manuel est contenu dans un fichier Zip, vous devez spécifier un chemin d'accès relatif au fichier Zip.
8. (Facultatif) Pour Extra Variables (Variables supplémentaires), entrez les variables que vous souhaitez que State Manager envoie à Ansible lors de l'exécution.
9. (Facultatif) Pour Check (Vérifier), choisissez une option.
10. (Facultatif) Pour Verbose (Détails), choisissez une option.
11. Pour Targets (Cibles), choisissez une option. Pour plus d'informations sur l'utilisation des cibles, consultez À propos des cibles et des contrôles du débit dans les associations State Manager (p. 990).
12. Dans la section Specify schedule (Spécifier une planification), choisissez On Schedule (Selon planification) ou No schedule (Pas de planification). Si vous choisissez On Schedule (Selon planification), utilisez les boutons fournis pour créer une planification de type cron ou rate pour l'association.
13. Dans la section Advanced options (Options avancées), pour Compliance severity (Gravité de conformité), choisissez un niveau de gravité pour l'association. Les rapports de conformité indiquent si l'état de l'association est conforme ou non conforme, ainsi que le niveau de gravité que vous spécifiez ici. Pour plus d'informations, consultez la section A propos de la conformité des associations State Manager (p. 783).

14. Dans la section Rate control (Contrôle du débit), configurez des options pour l'exécution d'associations State Manager dans un parc d'instances gérées. Pour plus d'informations sur l'utilisation des contrôles de débit, consultez À propos des cibles et des contrôles du débit dans les associations State Manager (p. 990).

    Dans la section Simultanéité, choisissez une option :

    - Choisissez targets (cibles) pour entrer un nombre absolu de cibles pouvant exécuter l'association simultanément.
    - Choisissez percentage (pourcentage) pour saisir un pourcentage de l'ensemble de cibles pouvant exécuter l'association simultanément.

    Dans la section Error threshold (Seuil d'erreurs), choisissez une option :

    - Choisissez errors (erreurs) pour saisir un nombre absolu d'erreurs autorisées avant que State Manager ne cesse d'exécuter des associations sur des cibles supplémentaires.
    - Choisissez percentage (pourcentage) pour saisir un pourcentage d'erreurs autorisées avant que State Manager ne cesse d'exécuter des associations sur des cibles supplémentaires.
15. (Facultatif) Dans Output options (Options de sortie), pour enregistrer la sortie de la commande dans un fichier, sélectionnez Enable writing to an S3 bucket (Autoriser l'écriture dans un compartiment S3) Entrez les noms de compartiment et de préfixe (dossier) dans les zones.

    > **Note**
    >
    > Les autorisations S3 qui accordent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour de plus amples informations, veuillez consulter Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). En outre, si le compartiment S3 spécifié se trouve dans un compte AWS différent, assurez-vous que le profil d'instance associé à l'instance dispose des autorisations nécessaires pour écrire dans ce compartiment.
16. Choisissez Create Association (Créer une association).

    > **Note**
    >
    > Si vous utilisez des balises pour créer une association sur une ou plusieurs instances cibles, puis que vous supprimez les balises d'une instance, cette instance n'exécute plus l'association. L'instance est dissociée du document State Manager.

Créer une association exécutant les manuels stratégiques Ansible (interface de ligne de commande)

La procédure suivante décrit comment utiliser l'AWS CLI pour créer une association State Manager qui exécute des manuels stratégiques Ansible à l'aide du document `AWS-ApplyAnsiblePlaybooks`.

Pour créer une association qui exécute des manuels stratégiques Ansible (interface de ligne de commande)

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).
2. Exécutez l'une des commandes suivantes pour créer une association qui exécute des manuels stratégiques Ansible en ciblant des instances à l'aide de balises Amazon EC2. La commande (A) spécifie GitHub comme type de source. La commande (B) spécifie Amazon S3 comme type de source.

   GitHubSource (A)

Linux

```
aws ssm create-association --name "AWS-ApplyAnsiblePlaybooks" \
--targets Key=tag:TagKey,Values=TagValue \
--parameters '{"SourceType":["GitHub"],"SourceInfo":
["{\"owner\":\"owner_name\", \"repository\": \"name\",
 \"getOptions\": \"branch:master\"}"],"InstallDependencies":
["True_or_False"],"PlaybookFile":["file_name.yml"],"ExtraVariables":["key/
value_pairs_separated_by_a_space"],"Check":["True_or_False"],"Verbose":["-v,-vv,-
vvv, or -vvvv"]}' \
--association-name "name" --schedule-expression "cron_or_rate_expression"
```

Windows

```
aws ssm create-association --name "AWS-ApplyAnsiblePlaybooks" ^
--targets Key=tag:TagKey,Values=TagValue ^
--parameters '{"SourceType":["GitHub"],"SourceInfo":
["{\"owner\":\"owner_name\", \"repository\": \"name\",
 \"getOptions\": \"branch:master\"}"],"InstallDependencies":
["True_or_False"],"PlaybookFile":["file_name.yml"],"ExtraVariables":["key/
value_pairs_separated_by_a_space"],"Check":["True_or_False"],"Verbose":["-v,-vv,-
vvv, or -vvvv"]}' ^
--association-name "name" --schedule-expression "cron_or_rate_expression"
```

Voici un exemple :

```
aws ssm create-association --name "AWS-ApplyAnsiblePlaybooks" \
--targets "Key=tag:OS,Values=Linux" \
--parameters '{"SourceType":["GitHub"],"SourceInfo":["{\"owner\":
\"ansibleDocumentTest\", \"repository\": \"Ansible\", \"getOptions\":
 \"branch:master\"}"],"InstallDependencies":["True"],"PlaybookFile":["hello-world-
playbook.yml"],"ExtraVariables":["SSM=True"],"Check":["False"],"Verbose":["-v"]}' \
--association-name "AnsibleAssociation" --schedule-expression "cron(0 2 ? * SUN *)"
```

Source S3 (B)

Linux

```
aws ssm create-association --name "AWS-ApplyAnsiblePlaybooks" \
--targets Key=tag:TagKey,Values=TagValue \
--parameters '{"SourceType":["S3"],"SourceInfo":["{\"path\":\"https://
s3.amazonaws.com/
path_to_Zip_file,_directory,_or_playbook_to_download\"}"],"InstallDependencies":
["True_or_False"],"PlaybookFile":["file_name.yml"],"ExtraVariables":["key/
value_pairs_separated_by_a_space"],"Check":["True_or_False"],"Verbose":["-v,-vv,-
vvv, or -vvvv"]}' \
--association-name "name" --schedule-expression "cron_or_rate_expression"
```

Windows

```
aws ssm create-association --name "AWS-ApplyAnsiblePlaybooks" ^
--targets Key=tag:TagKey,Values=TagValue ^
--parameters '{"SourceType":["S3"],"SourceInfo":["{\"path\":\"https://
s3.amazonaws.com/
path_to_Zip_file,_directory,_or_playbook_to_download\"}"],"InstallDependencies":
["True_or_False"],"PlaybookFile":["file_name.yml"],"ExtraVariables":["key/
value_pairs_separated_by_a_space"],"Check":["True_or_False"],"Verbose":["-v,-vv,-
vvv, or -vvvv"]}' ^
```

```
--association-name "name" --schedule-expression "cron_or_rate_expression"
```

Voici un exemple :

```
aws ssm create-association --name "AWS-ApplyAnsiblePlaybooks" \
--targets "Key=tag:OS,Values= Windows" \
--parameters '{"SourceType":["S3"],"SourceInfo":["{\"path\":\"https://s3.amazonaws.com/
DOC-EXAMPLE-BUCKET/playbook.yml\"}"],"InstallDependencies":["True"],"PlaybookFile":
["playbook.yml"],"ExtraVariables":["SSM=True"],"Check":["False"],"Verbose":["-v"]}' \
--association-name "AnsibleAssociation" --schedule-expression "cron(0 2 ? * SUN *)"
```

Note

Les associations State Manager ne prennent pas en charge toutes les expressions cron et rate. Pour plus d'informations sur la création d'expressions cron et rate pour des associations, consultez Référence : Expressions cron et rate pour Systems Manager (p. 1392).

Le système tente de créer l'association sur les instances et applique immédiatement l'état.

3. Exécutez la commande suivante pour afficher le statut mis à jour de l'association que vous venez de créer.

```
aws ssm describe-association --association-id "ID"
```

Création d'associations qui exécutent des recettes de Chef

Vous pouvez créer des associations State Manager qui exécutent des recettes Chef à l'aide du document `AWS-ApplyChefRecipes`. Vous pouvez cibler des nœuds gérés Systems Manager basés sur Linux avec le document `AWS-ApplyChefRecipes`. Ce document offre les avantages suivants pour l'exécution de recettes Chef :

- Il prend en charge plusieurs versions de Chef (Chef 11 à Chef 14).
- Il installe automatiquement le logiciel client Chef sur les instances cibles.
- Exécute éventuellement des contrôles de conformité Systems Manager (p. 778) sur les instances cibles et stocke les résultats des contrôles de conformité dans un compartiment S3.
- Il exécute plusieurs livres de cuisine et recettes en une seule fois du document.
- Il peut exécuter des recettes en mode `why-run`, pour afficher lesquelles changeront sur les instances cibles sans y apporter de modifications.
- Il peut appliquer des attributs JSON personnalisés aux exécutions `chef-client`.

Vous pouvez utiliser GitHub des compartiments S3 ou comme sources pour les livres de recettes et les recettes Chef que vous spécifiez dans un `AWS-ApplyChefRecipes` document .

Prérequis : Configurer votre association, votre référentiel et vos livres de recettes

Avant de créer un document `AWS-ApplyChefRecipes`, préparez votre répertoire de livres de recettes et de livres de recettes Chef. Si vous n'avez pas encore de livre de recettes Chef que vous souhaitez utiliser, vous pouvez commencer en utilisant un livre de recettes de test `HelloWorld` préparé pour vous par AWS. Le document `AWS-ApplyChefRecipes` pointe déjà vers ce livre de recettes par défaut. Vos livres de recettes doivent être configurés de la même manière que la structure de répertoire suivante. Dans l'exemple suivant, `jenkins` et `nginx` sont des exemples de livres de recettes Chef disponibles dans le Chef Supermarket sur le site web Chef.

Bien que AWS ne puissent pas officiellement prendre en charge les livres de recettes sur le site Chef Supermarket la plupart d'entre eux fonctionnent avec le document `AWS-ApplyChefRecipes`. Voici des exemples de critères à vérifier lorsque vous testez un livre de recette de la communauté :

- Le livre de recettes doit prendre en charge les systèmes d'exploitation basés sur Linux des nœuds gérés Systems Manager que vous ciblez.
- Le livre de recettes doit être valide pour la version client de Chef (Chef 11 à Chef 14) que vous utilisez.
- Le livre de recettes est compatible avec Chef Infra Client et ne nécessite pas de serveur Chef.

Vérifiez que vous pouvez accéder au site web Chef.io, afin que tous les livres de recettes que vous spécifiez dans votre liste d'exécution puissent être installés lors de l'exécution du document Systems Manager. L'utilisation d'un fichier `cookbooks` imbriqué est prise en charge, mais pas obligatoire ; vous pouvez stocker des livres de recettes directement dans le niveau racine.

```
<Top-level directory, or the top level of the archive file (ZIP or tgz or tar.gz)>
    ### cookbooks (optional level)
        ### jenkins
        #   ### metadata.rb
        #   ### recipes
        ### nginx
            ### metadata.rb
            ### recipes
```

> **Important**
>
> Avant de créer une State Manager association qui exécute les recettes Chef, sachez que l'exécution du document installe le logiciel client Chef sur vos nœuds Systems Manager gérés, sauf si vous définissez la valeur de la version du client Chef sur `None` . Cette action utilise un script d'installation de Chef pour installer les composants Chef en votre nom. Avant d'exécuter un document `AWS-ApplyChefRecipes`, assurez-vous que votre entreprise peut se conformer à toutes les exigences légales en vigueur, y compris les conditions de licence applicables à l'utilisation du logiciel Chef. (Pour plus d'informations, consultez le site web Chef.

Systems Manager peut fournir des rapports de conformité à un compartiment S3, à la console Systems Manager ou rendre les résultats de conformité disponibles en réponse aux commandes d'API Systems Manager. Pour exécuter des rapports de conformité Systems Manager, le profil d'instance attaché aux instances gérées Systems Manager doit disposer des autorisations d'écriture dans le compartiment S3. Le profil d'instance doit être autorisé à utiliser l'API Systems Manager`PutComplianceItem`. Pour de plus amples informations sur la conformité Systems Manager, veuillez consulter Conformité de la configuration AWS Systems Manager (p. 778).

Journalisation de l'exécution du document

Lorsque vous exécutez un document Systems Manager à l'aide d'une association State Manager, vous pouvez configurer l'association pour choisir la sortie de l'exécution du document, et vous pouvez envoyer la sortie vers Amazon S3 ou Amazon CloudWatch Logs. Pour faciliter le dépannage lorsqu'une association a terminé son exécution, vérifiez qu'elle est configurée pour écrire la sortie de commande dans un compartiment S3 ou CloudWatch Logs. Pour plus d'informations, consultez la section Créer une association (p. 993).

Utiliser GitHub comme source de livre de recettes

Le document `AWS-ApplyChefRecipes` utilise le plug-in `aws:downloadContent` (p. 1248) pour télécharger les livres de recettes. Pour télécharger le contenu depuis GitHub , spécifiez les informations relatives à votre GitHub référentiel au document au format JSON. Voici un exemple :

```
{
   "owner":"TestUser",
```

```
   "repository":"GitHubCookbookRepository",
   "path":"cookbooks/HelloWorld",
   "getOptions":"branch:master",
   "tokenInfo":"{{ssm-secure:secure-string-token}}"
}
```

Utiliser Amazon S3 comme source de livre de recettes

Vous pouvez également stocker et télécharger des livres de recettes Chef dans Amazon S3 sous la forme d'un fichier `.zip` ou `tar.gz` unique ou d'une structure de répertoires. Pour télécharger du contenu depuis Amazon S3, vous devez spécifier le chemin d'accès au fichier. Voici deux exemples :

Exemple 1 : Téléchargement d'un livre de recettes spécifique

```
{
   "path":"https://s3.amazonaws.com/chef-cookbooks/HelloWorld.zip"
}
```

Exemple 2 : Téléchargement du contenu d'un répertoire

```
{
   "path":"https://s3.amazonaws.com/chef-cookbooks-test/HelloWorld"
}
```

> **Important**
>
> Si vous spécifiez Amazon S3, le profil d'instance AWS Identity and Access Management (IAM) sur vos instances gérées doit être configuré avec la stratégie `AmazonS3ReadOnlyAccess`. Pour plus d'informations, consultez la section Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34).

Rubriques

- Créer une association qui exécute les recettes de Chef (console) (p. 1037)
- Créer une association qui exécute des recettes de Chef (CLI) (p. 1039)
- Affichage des détails de conformité des ressources Chef (p. 1041)

Créer une association qui exécute les recettes de Chef (console)

La procédure suivante décrit comment utiliser la console Systems Manager pour créer une association State Manager qui exécute des livres de recettes Chef à l'aide du document `AWS-ApplyChefRecipes`.

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez State Manager, puis Créer une association.
3. Pour Name (Nom), saisissez un nom qui vous aide à mémoriser l'objectif de l'association.
4. Dans la liste Document, choisissez AWS-ApplyChefRecipes.
5. Dans Parameters (Paramètres), pour Source Type (Type de source), choisissez GitHub ou S3.
6. Dans Source info (Informations sur la source), saisissez les informations sur la source du livre de recettes dans l'un des formats suivants.

   a. Si vous avez choisi GitHub à l'étape 5, entrez les informations de référentiel au format suivant :

   ```
   {
      "owner":"user_name",
      "repository":"name",
   ```

```
    "path":"path_to_directory_or_cookbook_to_download",
    "getOptions":"branch:branch_name",
    "tokenInfo":"{{(Optional)_token_information}}"
}
```

b. Si vous avez choisi S3 à l'étape 5, saisissez les informations de chemin au format suivant :

```
{
    "path":"https://s3.amazonaws.com/path_to_directory_or_cookbook_to_download"
}
```

7. Dans la Run list (Liste d'exécution), listez les recettes que vous souhaitez exécuter au format suivant, en séparant chaque recette par une virgule comme indiqué. N'ajoutez pas d'espace après la virgule.

```
recipe[cookbook_name1::recipe_name],recipe[cookbook_name2::recipe_name]
```

8. (Facultatif) Dans le JSON attributes content (Contenu des attributs JSON), ajoutez tout JSON personnalisé qui contient des attributs que le client Chef doit transmettre à vos instances cibles.

   Le paramètre JSON attributes content (Contenu des attributs JSON) est plus utile aux fins suivantes :

   - Lorsque vous voulez remplacer uniquement un petit nombre d'attributs et vous n'avez pas besoin d'utiliser les livres personnalisés dans d'autres buts.

     JSON personnalisé peut vous aider à éviter le travail supplémentaire de configuration et de maintenance d'un référentiel de livres de recettes pour remplacer seulement quelques attributs.
   - Valeurs qui sont censées varier.

     Par exemple, si vos livres de recettes Chef configurent une application tierce qui accepte les paiements, vous pouvez utiliser JSON personnalisé pour spécifier l'URL du point de terminaison de paiement. Si le fabricant du logiciel tiers modifie l'URL du point de terminaison de paiement, vous pouvez utiliser JSON personnalisé pour mettre à jour le point de terminaison de paiement vers la nouvelle URL.
9. Pour la Chef client version (Version du client Chef), spécifiez une version de Chef. Les valeurs valides sont `11` , `12` , `13` , ou `14` .`None` Si vous spécifiez `11` via `14` , Systems Manager installe la version du client Chef correcte sur vos instances cibles. Si vous spécifiez `None`, Systems Manager n'installe pas le client Chef sur les instances cibles avant d'exécuter les recettes du document. La valeur par défaut est `14`.
10. (Facultatif) Pour les Chef client arguments (Arguments du client Chef), spécifiez des arguments supplémentaires pris en charge pour la version de Chef que vous utilisez. Pour en savoir plus sur les arguments pris en charge, exécutez `chef-client -h` sur une instance qui exécute le client Chef.
11. (Facultatif) Activez Why-run pour afficher les modifications qui seront apportées aux instances cibles si les recettes sont exécutées, sans modifier réellement les instances cibles.
12. Pour Compliance severity (Gravité de conformité), choisissez la gravité des résultats de conformité de la configuration Systems Manager que vous souhaitez rapportés. Les rapports de conformité indiquent si l'état de l'association est conforme ou non conforme, ainsi que le niveau de gravité que vous spécifiez. Les rapports de conformité de la configuration sont stockés dans un comp)artiment S3 que vous spécifiez comme valeur du paramètre de Compliance report bucket (Compartiment de rapport de conformité (étape 15). Pour de plus amples informations sur la conformité de la configuration, veuillez consulter Utilisation de la conformité de configuration (p. 782) dans ce guide.

    Les analyses de conformité mesurent l'écart entre la configuration spécifiée dans vos recettes Chef et les ressources d'instance. Les valeurs valides sont `Critical` , `High` , `Medium` , `Low` , , ou `Informational` .`UnspecifiedNone` Pour ignorer les rapports de conformité, choisissez `None` .
13. Pour Compliance type (Type de conformité), spécifiez le type de conformité pour lequel vous souhaitez que les résultats soient rapportés. Les valeurs valides sont `Association` pour les State Manager associations ou`Custom:`*`custom_type`*. La valeur par défaut est `Custom:Chef` .

14. Pour le Compliance report bucket (Compartiment de rapport de conformité), saisissez le nom d'un compartiment S3 dans lequel stocker des informations sur chaque Chef exécuté par ce document, y compris la configuration des ressources et les résultats de la conformité de la configuration.
15. Dans Rate control (Contrôle du débit), configurez des options pour l'exécution d'associations State Manager dans un parc d'instances gérées. Pour plus d'informations sur l'utilisation des contrôles de débit, consultez À propos des cibles et des contrôles du débit dans les associations State Manager (p. 990).

    Dans Concurrency (Simultanéité), choisissez une option :

    - Choisissez targets (cibles) pour entrer un nombre absolu de cibles pouvant exécuter l'association simultanément.
    - Choisissez percentage (pourcentage) pour saisir un pourcentage de l'ensemble de cibles pouvant exécuter l'association simultanément.

    Dans Error threshold (Seuil d'erreur), choisissez une option :

    - Choisissez errors (erreurs) pour saisir un nombre absolu d'erreurs autorisées avant que State Manager ne cesse d'exécuter des associations sur des cibles supplémentaires.
    - Choisissez percentage (pourcentage) pour saisir un pourcentage d'erreurs autorisées avant que State Manager ne cesse d'exécuter des associations sur des cibles supplémentaires.
16. (Facultatif) Dans Output options (Options de sortie), pour enregistrer la sortie de la commande dans un fichier, sélectionnez Enable writing to an S3 bucket (Autoriser l'écriture dans un compartiment S3) Entrez les noms de compartiment et de préfixe (dossier) dans les zones.

    > **Note**
    >
    > Les autorisations S3 qui accordent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour de plus amples informations, veuillez consulter Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). En outre, si le compartiment S3 spécifié se trouve dans un compte AWS différent, assurez-vous que le profil d'instance associé à l'instance dispose des autorisations nécessaires pour écrire dans ce compartiment.
17. Choisissez Create Association (Créer une association).

Créer une association qui exécute des recettes de Chef (CLI)

La procédure suivante décrit comment utiliser l'AWS CLI pour créer une association State Manager qui exécute des livres de recettes Chef à l'aide du document `AWS-ApplyChefRecipes`.

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).
2. Exécutez l'une des commandes suivantes pour créer une association qui exécute des livres de recettes Chef en ciblant des instances à l'aide de balises Amazon EC2. La commande (A) utilise GitHub comme type de source. La commande (B) utilise Amazon S3 comme type de source.

   GitHubSource (A)

   Linux

```
aws ssm create-association --name "AWS-ApplyChefRecipes" \
--targets Key=tag:TagKey,Values=TagValue \
--parameters '{"SourceType":["GitHub"],"SourceInfo":["{\"owner
\":\"owner_name\", \"repository\": \"name\", \"path\":
```

```
 \"path_to_directory_or_cookbook_to_download\", \"getOptions\":
 \"branch:branch_name\"}"], "RunList":["{\"recipe[cookbook_name1::recipe_name]\",
 \"recipe[cookbook_name2::recipe_name]\"}"], "JsonAttributesContent":
 ["{Custom_JSON}"], "ChefClientVersion": ["version_number"], "ChefClientArguments":
["{chef_client_arguments}"], "WhyRun": true_or_false, "ComplianceSeverity":
 ["severity_value"], "ComplianceType": ["Custom:Chef"], "ComplianceReportBucket":
 ["DOC-EXAMPLE-BUCKET"]}' \
--association-name "name" --schedule-expression "cron_or_rate_expression"
```

Windows

```
aws ssm create-association --name "AWS-ApplyChefRecipes" ^
--targets Key=tag:TagKey,Values=TagValue ^
--parameters '{"SourceType":["GitHub"],"SourceInfo":["{\"owner
\":\"owner_name\", \"repository\": \"name\", \"path\":
 \"path_to_directory_or_cookbook_to_download\", \"getOptions\":
 \"branch:branch_name\"}"], "RunList":["{\"recipe[cookbook_name1::recipe_name]\",
 \"recipe[cookbook_name2::recipe_name]\"}"], "JsonAttributesContent":
 ["{Custom_JSON}"], "ChefClientVersion": ["version_number"], "ChefClientArguments":
["{chef_client_arguments}"], "WhyRun": true_or_false, "ComplianceSeverity":
 ["severity_value"], "ComplianceType": ["Custom:Chef"], "ComplianceReportBucket":
 ["DOC-EXAMPLE-BUCKET"]}' ^
--association-name "name" --schedule-expression "cron_or_rate_expression"
```

Voici un exemple :

```
aws ssm create-association --name "AWS-ApplyChefRecipes" \
--targets Key=tag:OS,Values=Linux \
--parameters '{"SourceType":["GitHub"],"SourceInfo":["{\"owner\":\"ChefRecipeTest\",
 \"repository\": \"ChefCookbooks\", \"path\": \"cookbooks/HelloWorld\", \"getOptions
\": \"branch:master\"}"], "RunList":["{\"recipe[HelloWorld::HelloWorldRecipe]\",
 \"recipe[HelloWorld::InstallApp]\"}"], "JsonAttributesContent": ["{\"state\":
 \"visible\",\"colors\": {\"foreground\": \"light-blue\",\"background\": \"dark-gray
\"}}"], "ChefClientVersion": ["14"], "ChefClientArguments":["{--fips}"], "WhyRun":
 false, "ComplianceSeverity": ["Medium"], "ComplianceType": ["Custom:Chef"],
 "ComplianceReportBucket": ["ChefComplianceResultsBucket"]}' \
--association-name "MyChefAssociation" --schedule-expression "cron(0 2 ? * SUN *)"
```

Source S3 (B)

Linux

```
aws ssm create-association --name "AWS-ApplyChefRecipes" \
--targets Key=tag:TagKey,Values=TagValue \
--parameters '{"SourceType":["S3"],"SourceInfo":["{\"path\":\"https://
s3.amazonaws.com/path_to_Zip_file,_directory,_or_cookbook_to_download\"}"],
 "RunList":["{\"recipe[cookbook_name1::recipe_name]\",
 \"recipe[cookbook_name2::recipe_name]\"}"], "JsonAttributesContent":
 ["{Custom_JSON}"], "ChefClientVersion": ["version_number"], "ChefClientArguments":
["{chef_client_arguments}"], "WhyRun": true_or_false, "ComplianceSeverity":
 ["severity_value"], "ComplianceType": ["Custom:Chef"], "ComplianceReportBucket":
 ["DOC-EXAMPLE-BUCKET"]}' \
--association-name "name" --schedule-expression "cron_or_rate_expression"
```

Windows

```
aws ssm create-association --name "AWS-ApplyChefRecipes" ^
--targets Key=tag:TagKey,Values=TagValue ^
--parameters '{"SourceType":["S3"],"SourceInfo":["{\"path\":\"https://
s3.amazonaws.com/path_to_Zip_file,_directory,_or_cookbook_to_download\"}"],
```

```
 "RunList":["{\"recipe[cookbook_name1::recipe_name]\",
 \"recipe[cookbook_name2::recipe_name]\"}"], "JsonAttributesContent":
 ["{Custom_JSON}"], "ChefClientVersion": ["version_number"], "ChefClientArguments":
["{chef_client_arguments}"], "WhyRun": true_or_false, "ComplianceSeverity":
 ["severity_value"], "ComplianceType": ["Custom:Chef"], "ComplianceReportBucket":
 ["DOC-EXAMPLE-BUCKET"]}' ^
--association-name "name" --schedule-expression "cron_or_rate_expression"
```

Voici un exemple :

```
aws ssm create-association --name "AWS-ApplyChefRecipes" ^
--targets "Key=tag:OS,Values= Windows" ^
--parameters '{"SourceType":["S3"],"SourceInfo":["{\"path\":\"https://
s3.amazonaws.com/DOC-EXAMPLE-BUCKET/HelloWorld\"}"], "RunList":
["{\"recipe[HelloWorld::HelloWorldRecipe]\", \"recipe[HelloWorld::InstallApp]\"}"],
 "JsonAttributesContent": ["{\"state\": \"visible\",\"colors\": {\"foreground\":
 \"light-blue\",\"background\": \"dark-gray\"}}"], "ChefClientVersion": ["14"],
 "ChefClientArguments":["{--fips}"], "WhyRun": false, "ComplianceSeverity":
 ["Medium"], "ComplianceType": ["Custom:Chef"], "ComplianceReportBucket":
 ["ChefComplianceResultsBucket"]}' ^
--association-name "name" --schedule-expression "cron(0 2 ? * SUN *)"
```

> **Note**
>
> Les associations State Manager ne prennent pas en charge toutes les expressions cron et rate. Pour plus d'informations sur la création d'expressions cron et rate pour des associations, consultez Référence : Expressions cron et rate pour Systems Manager (p. 1392).

Le système tente de créer l'association sur les instances et applique immédiatement l'état.

3. Exécutez la commande suivante pour afficher le statut mis à jour de l'association que vous venez de créer.

```
aws ssm describe-association --association-id "ID"
```

Affichage des détails de conformité des ressources Chef

Systems Manager capture les informations de conformité sur les ressources gérées par Chef dans la valeur Amazon Simple Storage Service (Amazon S3) Compliance report bucket (Compartiment de rapport de conformité) que vous avez spécifiée lors de l'exécution du document `AWS-ApplyChefRecipes`. La recherche d'informations sur les défaillances de ressources Chef dans un compartiment S3 peut prendre du temps. Au lieu de cela, vous pouvez consulter ces informations sur la page Systems Manager Compliance (Conformité).

Une analyse de conformité Systems Manager collecte des informations sur les ressources de vos nœuds gérés qui ont été créées ou vérifiées lors de la dernière exécution de Chef. Les ressources peuvent inclure des fichiers, des répertoires, des services `systemd`, des packages `yum`, des fichiers modélisés, des packages `gem` et des livres de recettes dépendants, entre autres.

La section Récapitulatif des ressources de conformité affiche le nombre de ressources qui ont échoué. Dans l'exemple suivant, le ComplianceType est personnalisé : Chef et une ressource est non conforme.

> **Note**
>
> `Custom:Chef`est la ComplianceType valeur par défaut dans le `AWS-ApplyChefRecipes` document. Cette valeur est personnalisable.

Compliance resources summary

Compliance type	Compliant resources	Non-Compliant resources	Critical resources	High resources	Medium resources	Low resources
Custom:Chef	1	0	0	0	0	0

La section Details overview for resources (Présentation détaillée des ressources) montre des informations sur la ressource AWS qui n'est pas conforme. Cette section inclut également le type de ressource Chef par rapport auquel la conformité a été exécutée, la gravité du problème, l'état de conformité et des liens pour en savoir plus, le cas échéant.

Details overview for resources

Resource

ID	Resource type	Compliance type	Overall severity	Overall status
i-0[redacted]6	ManagedInstance	Custom:Chef	Critical	Compliant

Compliance rule

All

Status : Equal : Compliant | ComplianceType : Equal : Custom:Chef | Severity : Equal : All | ResourceId : Equal : i-0[redacted]

ID	Compliance type	Resource ID	Severity	Status
aws-site::install-nginx::nginx	Custom:Chef	i-0[redacted]6	Critical	Compliant
aws-site::install-nginx::nginx	Custom:Chef	i-0[redacted]6	Critical	Compliant
aws-site::install-nginx::/var/www/html/	Custom:Chef	i-0[redacted]6	Critical	Compliant
aws-site::install-nginx::/etc/nginx/nginx.conf	Custom:Chef	i-0[redacted]6	Critical	Compliant
aws-site::deploy-app::/usr/share/nginx/html/index.html	Custom:Chef	i-0[redacted]6	Critical	Compliant

La View output (Afficher la sortie) affiche les 4 000 derniers caractères du statut détaillé. Systems Manager commence par l'exception comme premier élément, puis trouve les messages détaillés et en affiche autant

que possible jusqu'à la limite de 4 000 caractères. Ce processus affiche les messages de journal qui ont été générés avant que l'exception soit déclenchée, qui sont les plus pertinents pour la résolution.

Pour de plus amples informations sur la façon d'afficher les informations de conformité, veuillez consulter Conformité de la configuration AWS Systems Manager (p. 778).

Les échecs d'association affectent les rapports de conformité

Si l'association State Manager échoue, aucune donnée de conformité n'est présentée. Par exemple, si Systems Manager tente de télécharger un livre de recettes à partir d'un compartiment S3 auquel l'instance n'est pas autorisée à accéder, l'association échoue et aucune donnée de conformité n'est présentée par Systems Manager.

Mise à jour automatique d'Agent SSM (interface de ligne de commande)

La procédure suivante vous guide au cours du processus de création d'une association State Manager à l'aide de l'AWS Command Line Interface (AWS CLI). L'association met automatiquement à jour Agent SSM selon une planification que vous spécifiez. Pour plus d'informations sur l'élément Agent SSM, consultez la section Utilisation des Agent SSM (p. 72).

> **Note**
>
> Notez les détails suivants sur la mise à jour automatique Agent SSM d' :
>
> - À compter du 21 septembre 2020, la mise à jour automatique installe Agent SSM version 3.0. Pour de plus amples informations, veuillez consulter Agent SSMversion 3 (p. 75).
> - Pour recevoir des notifications sur les Agent SSM mises à jour, abonnez-vous à la page Agent SSM Notes de mise à jour sur GitHub .

Avant de commencer

Avant de réaliser la procédure suivante, veillez à avoir au moins une instance EC2 en cours d'exécution pour Linux ou Windows Server, configurée pour Systems Manager. Pour plus d'informations, consultez la section Prérequis d'Systems Manager (p. 15).

> **Note**
>
> Si vous créez une association en utilisant AWS CLI ou Outils AWS pour Windows PowerShell, utilisez le paramètre `--Targets` pour les instances cibles, comme illustré à l'exemple suivant. N'utilisez pas le paramètre `--InstanceID`. Le paramètre `--InstanceID` est un paramètre hérité.

Pour créer une association afin de mettre à jour automatiquement Agent SSM

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).
2. Exécutez la commande suivante pour créer une association en ciblant les instances à l'aide des balises Amazon EC2. Le paramètre `Schedule` définit une planification pour exécuter l'association tous les dimanches matin à 2 h 00 (UTC).

   Linux

```
aws ssm create-association \
--targets Key=tag:tag_key,Values=tag_value \
--name AWS-UpdateSSMAgent \
--schedule-expression "cron(0 2 ? * SUN *)"
```

Windows

```
aws ssm create-association ^
--targets Key=tag:tag_key,Values=tag_value ^
--name AWS-UpdateSSMAgent ^
--schedule-expression "cron(0 2 ? * SUN *)"
```

> **Note**
>
> Les associations State Manager ne prennent pas en charge toutes les expressions cron et rate. Pour plus d'informations sur la création d'expressions cron et rate pour des associations, consultez Référence : Expressions cron et rate pour Systems Manager (p. 1392).

Si vous le souhaitez, vous pouvez également cibler plusieurs instances en spécifiant les instances IDs dans une liste séparée par des virgules.

Linux

```
aws ssm create-association \
--targets Key=instanceids,Values=instance_ID,instance_ID,instance_ID \
--name document_name \
--schedule-expression "cron(0 2 ? * SUN *)"
```

Windows

```
aws ssm create-association ^
--targets Key=instanceids,Values=instance_ID,instance_ID,instance_ID ^
--name document_name ^
--schedule-expression "cron(0 2 ? * SUN *)"
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
    "AssociationDescription": {
        "ScheduleExpression": "cron(0 2 ? * SUN *)",
        "Name": "AWS-UpdateSSMAgent",
        "Overview": {
            "Status": "Pending",
            "DetailedStatus": "Creating"
        },
        "AssociationId": "123.............",
        "DocumentVersion": "$DEFAULT",
        "LastUpdateAssociationDate": 1504034257.98,
        "Date": 1504034257.98,
        "AssociationVersion": "1",
        "Targets": [
            {
                "Values": [
                    "TagValue"
                ],
                "Key": "tag:TagKey"
            }
        ]
    }
}
```

Le système tente de créer l'association sur les instances et applique immédiatement l'état. Le statut de l'association indique `Pending`.

3. Exécutez la commande suivante pour afficher le statut mis à jour de l'association que vous venez de créer.

```
aws ssm list-associations
```

> **Note**
>
> Si vos instances n'exécutent pas la version la plus récente de l' Agent SSM , le statut indique `Failed` . Il s'agit du comportement attendu. Lors de la publication d'une nouvelle version d'Agent SSM, l'association installe automatiquement le nouvel agent et le statut indique `Success`.

Procédure : Mise à jour automatique des pilotes PV sur les instances EC2 pour Windows Server (console)

Amazon Windows AMIs contient un ensemble de pilotes pour permettre l'accès au matériel virtualisé. Ces pilotes sont utilisés par Amazon Elastic Compute Cloud (Amazon EC2) pour mapper le stockage d'instance et les volumes Amazon Elastic Block Store (Amazon EBS) à leurs périphériques. Nous vous recommandons d'installer les derniers pilotes pour améliorer la stabilité et les performances des instances EC2 pour Windows Server. Pour plus d'informations sur les pilotes PV, consultez Pilotes PV AWS.

La procédure suivante explique comment configurer une association State Manager pour télécharger et installer automatiquement de nouveaux pilotes PV AWS dès leur mise à disposition.

Avant de commencer

Avant d'exécuter la procédure suivante, veillez à avoir au moins une instance EC2 pour Windows Server en cours d'exécution et configurée pour Systems Manager. Pour plus d'informations, consultez la section Prérequis d'Systems Manager (p. 15).

Pour créer une association State Manager qui met automatiquement à jour les pilotes PV

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez State Manager.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez State Manager.
3. Choisissez Créer une association.
4. Dans le champ Nom, entrez un nom descriptif.
5. Dans la liste Document, choisissez AWS-ConfigureAWSPackage.
6. Dans la section Paramètres, choisissez Installer dans la liste Action.
7. Pour Installation type (Type d'installation), choisissez l'une des valeurs suivantes :

   - Désinstaller et réinstaller : Le package est complètement désinstallé, puis réinstallé. L'application n'est pas disponible tant que la réinstallation n'est pas terminée.
   - Mise à jour sur place : Seuls les fichiers nouveaux ou modifiés sont ajoutés à l'installation existante selon les instructions que vous fournissez dans un `update` script . L'application reste disponible tout au long du processus de mise à jour. Cette option n'est pas prise en charge pour les packages publiés par .AWS

8. Dans le champ Nom, entrez AWSPVDriver . Vous pouvez laisser les champs Version et Additional Arguments vides.
9. Dans la section Targets (Cibles), choisissez une option.

   Note

   Si vous choisissez de cibler les instances à l'aide de balises et que vous spécifiez des balises qui correspondent aux instances Linux, l'association réussit sur l'instance Windows, mais échoue sur les instances Linux. Le statut global de l'association indique Failed.
10. Dans la section Spécifier le programme, choisissez une option. Comme des pilotes PV mis à jour sont publiés à plusieurs reprises dans l'année, vous pouvez planifier l'association pour qu'elle s'exécute une fois par mois, si vous le désirez.
11. Dans la section Advanced options (Options avancées), pour Compliance severity (Gravité de conformité), choisissez un niveau de gravité pour l'association. Les rapports de conformité indiquent si l'état de l'association est conforme ou non conforme, ainsi que le niveau de gravité que vous spécifiez ici. Pour plus d'informations, consultez la section A propos de la conformité des associations State Manager (p. 783).
12. (Facultatif) Pour Contrôle du débit :
    - Pour Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage d'instances sur lesquelles exécuter simultanément la commande.

      Note

      Si vous avez sélectionné des cibles en indiquant des balises appliquées à des instances gérées ou en indiquant des groupes de ressources AWS, et que vous n'êtes pas sûr du nombre d'instances ciblées, limitez le nombre d'instances pouvant exécuter simultanément le document en spécifiant un pourcentage.
    - Pour Seuil d'erreur, spécifiez quand arrêter l'exécution de la commande sur d'autres instances après son échec sur un nombre ou un pourcentage d'instances. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les instances sur lesquelles la commande est en cours de traitement peuvent également envoyer des erreurs.
13. (Facultatif) Dans Output options (Options de sortie), pour enregistrer la sortie de la commande dans un fichier, sélectionnez Enable writing to an S3 bucket (Autoriser l'écriture dans un compartiment S3) Entrez les noms de compartiment et de préfixe (dossier) dans les zones.

    Note

    Les autorisations S3 qui accordent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour de plus amples informations, veuillez consulter Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). En outre, si le compartiment S3 spécifié se trouve dans un compte AWS différent, assurez-vous que le profil d'instance associé à l'instance dispose des autorisations nécessaires pour écrire dans ce compartiment.
14. Choisissez Créer une association, puis Fermer. Le système tente de créer l'association sur les instances et applique immédiatement l'état.

    Si vous avez créé l'association sur une ou plusieurs instances EC2 pour Windows Server, le statut se change en Success. Si vos instances ne sont pas correctement configurées pour Systems Manager ou si vous avez malencontreusement ciblé des instances Linux, le statut indique Failed (Échec).

    Si le statut est Failed, choisissez l'ID d'association, choisissez l'onglet Resources, puis vérifiez que l'association a été créée avec succès sur vos instances EC2 pour Windows Server . Si les instances EC2 pour Windows Server affichent le statut Failed (Échec), vérifiez que l'Agent SSM s'exécute sur l'instance et que celle-ci est configurée avec un rôle IAM pour Systems Manager. Pour plus d'informations, consultez la section Prérequis d'Systems Manager (p. 15).

AWS Systems Manager Gestionnaire de correctifs

AWS Systems ManagerGestionnaire de correctifsautomatise le processus d'application des correctifs des instances gérées avec les mises à jour liées à la sécurité et autres. Vous pouvez utiliser Gestionnaire de correctifs pour appliquer des correctifs pour les systèmes d'exploitation et les applications. (Sur Windows Server , la prise en charge des applications est limitée aux mises à jour des applications Microsoft.) Vous pouvez utiliser Gestionnaire de correctifs pour installer les Service Packs sur les instances Windows et effectuer des mises à niveau de version mineure sur les instances Linux. Vous pouvez corriger les parcs d'instances EC2 ou vos serveurs sur site et machines virtuelles (VM) par type de système d'exploitation. Cela inclut les versions prises en charge d' Windows Server , Amazon Linux , Amazon Linux 2 , CentOS , Debian Server , , Oracle Linux (Red Hat Enterprise Linux), RHEL (SUSE Linux Enterprise Server) et SLES .Ubuntu Server Vous pouvez scanner les instances pour afficher uniquement le rapport des correctifs manquants, ou scanner et installer automatiquement les correctifs manquants.

> **Important**
>
> AWSne teste pas les correctifs pour Windows Server ou Linux avant leur mise à disposition dans Gestionnaire de correctifs . Gestionnaire de correctifsDe plus, ne prend pas en charge la mise à niveau des versions majeures des systèmes d'exploitation, telles que Windows Server 2016 vers Windows Server 2019, ou SUSE Linux Enterprise Server (SLES) 12.0 vers SLES 15.0.

Gestionnaire de correctifsutilise des références de correctifs, qui incluent les règles d'approbation automatique des correctifs quelques jours après leur publication, ainsi qu'une liste des correctifs approuvés et rejetés. Vous pouvez installer des correctifs régulièrement en planifiant leur exécution sous forme de tâche d'une fenêtre de maintenance de Systems Manager. Vous pouvez également installer les correctifs individuellement ou sur des groupes d'instances volumineux en utilisant des balises Amazon EC2. (Les balises sont des clés qui facilitent l'identification et le tri de vos ressources au sein de votre organisation.) Vous pouvez ajouter des balises à vos références de correctifs elles-mêmes lorsque vous les créez ou les mettez à jour.

Gestionnaire de correctifsfournit des options pour analyser vos instances et signaler la conformité selon un calendrier, installer les correctifs disponibles selon un calendrier, et corriger ou analyser les instances à la demande chaque fois que vous en avez besoin.

Gestionnaire de correctifss'intègre à AWS Identity and Access Management (IAM), et AWS CloudTrail pour fournir une expérience d'application de correctifs sécurisée qui inclut des notifications d'événements et la possibilité d'effectuer un audit de l'utilisation.Amazon EventBridge

Pour de plus amples informations sur l'utilisation de CloudTrail pour surveiller les actions Systems Manager, veuillez consulter Journalisation des appels d'API AWS Systems Manager avec AWS CloudTrail (p. 1336).

Pour de plus amples informations sur l'utilisation de EventBridge pour surveiller les événements Systems Manager, veuillez consulter Surveillance des événements Systems Manager avec Amazon EventBridge (p. 1340).

Rubriques

- Prérequis d'Gestionnaire de correctifs (p. 1048)
- Fonctionnement Gestionnaire de correctifs des opérations (p. 1049)
- À propos des opérations d'application de correctifs (p. 1076)
- À propos des références de correctifs (p. 1098)
- Correction d'instances hors conformité (Gestionnaire de correctifs) (p. 1110)
- Utilisation de Kernel Live Patching sur les instances Amazon Linux 2 (p. 1117)
- Utilisation d' Gestionnaire de correctifs (console) (p. 1123)
- Utilisation dGestionnaire de correctifs' AWS CLI () (p. 1147)
- Procédures Gestionnaire de correctifs AWS Systems Manager (p. 1166)

Prérequis d'Gestionnaire de correctifs

Version Agent SSM

Les versions 2.0.834.0 ou ultérieures d' Agent SSM s'exécutent sur les instances que vous souhaitez gérer avec Gestionnaire de correctifs .

> **Note**
>
> Une version mise à jour de l'Agent SSM est publiée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Si une version antérieure de l'agent est en cours d'exécution sur une instance, certains processus de l'Agent SSM peuvent échouer. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir l'Agent SSM à jour sur vos instances. Pour de plus amples informations, veuillez consulter Automatisation des mises à jour versAgent SSM (p. 109). Pour être alerté des mises à jour de Agent SSM, abonnez-vous à la page Agent SSM Release Notes sur GitHub.

Systèmes d'exploitation pris en charge

La Gestionnaire de correctifs fonctionnalité ne prend pas en charge les mêmes versions des systèmes d'exploitation qui sont prises en charge par d'autres AWS Systems Manager fonctionnalités. Par exemple, Gestionnaire de correctifs ne prend pas en charge CentOS 6.3 ou Raspbian Stretch. (Pour obtenir la liste complète des systèmes d'exploitation pris en charge par Systems Manager , consultez .)Prérequis d'Systems Manager (p. 15) Par conséquent, assurez-vous que les instances que vous souhaitez utiliser avec exécuteGestionnaire de correctifsnt l'un des systèmes d'exploitation répertoriés dans le tableau suivant.

Système d'exploitation	Details (Détails)
Linux	• Amazon Linux 2012.03 à 2018.03 • Amazon Linux 2 2 à 2.0 • CentOS6,5 - 7.8, 8,0-8.1 • Debian Server8.x et 9.x • Oracle Linux: 7,5 à 7.8 • Red Hat Enterprise Linux (RHEL) 6.5 à 8.2 • SUSE Linux Enterprise Server (SLES) 12.0 et versions ultérieures 12.x, 15.0 et 15.1 • Ubuntu Server14.04 LTS, 16.04 LTS, 18.04 LTS et 20.04 LTS **Note** Les instances créées depuis une AMI Amazon Linux qui utilisent un proxy doivent exécuter une version actuelle du module Python `requests` afin de prendre en charge les opérations du Gestionnaire de correctifs. Pour de plus amples informations, veuillez consulter Mettre à niveau le module de demandes Python sur Amazon Linux les instances qui utilisent un serveur proxy (p. 98).

Système d'exploitation	Details (Détails)
Windows	Windows Server 2008 à Windows Server 2019, y compris les versions R2.

Fonctionnement Gestionnaire de correctifs des opérations

Cette section fournit des détails techniques qui expliquent comment Gestionnaire de correctifs détermine les correctifs à installer et comment il les installe sur chaque système d'exploitation pris en charge. Pour les systèmes d'exploitation Linux, elle fournit également des informations sur la spécification d'un référentiel source pour les correctifs différent de celui par défaut configuré sur une instance, dans une référence de correctifs personnalisée. Cette section fournit également des détails sur le fonctionnement des règles de référence de correctifs sur différentes distributions du système d'exploitation Linux.

Les informations contenues dans les rubriques suivantes s'appliquent lorsque vous appliquez des correctifs à des instances selon une planification et à la demande.

Rubriques

- Sélection des correctifs de sécurité (p. 1049)
- Spécification d'un autre référentiel source de correctifs (Linux) (p. 1054)
- Installation des correctifs (p. 1056)
- Fonctionnement des règles de référence de correctif sur les systèmes basés sur Linux (p. 1063)
- Différences clés entre l'application de correctifs dans Windows et Linux (p. 1075)

Sélection des correctifs de sécurité

est principalement axé sur l'installation des mises à jour liées à la sécurité des systèmes d'exploitation sur les instances.Gestionnaire de correctifs Par défaut, Gestionnaire de correctifs n'installe pas tous les correctifs disponibles, mais plutôt un plus petit ensemble de correctifs axés sur la sécurité.

> Note
>
> Sur tous les systèmes Linux pris en charge par Gestionnaire de correctifs , vous pouvez choisir un autre référentiel source configuré pour l'instance, généralement pour installer des mises à jour non liées à la sécurité. Pour plus d'informations, consultez la section Spécification d'un autre référentiel source de correctifs (Linux) (p. 1054).

Le reste de cette section explique comment Gestionnaire de correctifs sélectionne les correctifs de sécurité pour les différents systèmes d'exploitation pris en charge.

Amazon Linux and Amazon Linux 2

Sur Amazon Linux et Amazon Linux 2, le service de référence de correctifs Systems Manager utilise des référentiels préconfigurés sur l'instance. Une instance comporte généralement deux référentiels (repos) préconfigurés :

- ID du référentiel : amzn-main/latest

  Nom du référentiel : amzn-main-Base
- ID du référentiel : amzn-updates/latest

  Nom du référentiel : amzn-updates-Base

> **Note**
>
> Toutes les mises à jour sont téléchargées à partir des référentiels distants configurés sur l'instance. Par conséquent, l'instance doit être en mesure de se connecter aux référentiels afin que l'application de correctifs s'exécute.

Amazon LinuxLes Amazon Linux 2 instances et utilisent Yum comme gestionnaire de package, et Yum utilise le concept de notice de mise à jour sous la forme d'un fichier nommé `updateinfo.xml` . Une notice de mise à jour est simplement un ensemble de packages qui corrigent des problèmes spécifiques. Tous les packages qui sont inclus dans une notice de mise à jour sont considérés comme des packages de sécurité par Gestionnaire de correctifs . Les packages ne se voient pas attribuer des classifications ou des niveaux de gravité. Pour cette raison, Gestionnaire de correctifs attribue les attributs d'une notice de mise à jour aux packages associés.

> **Note**
>
> Si vous cochez la case Approved patches include non-security updates (Les correctifs approuvés incluent des mises à jour non liées à la sécurité) dans la page Create patch baseline (Créer une référence de correctif), les packages qui ne sont pas classées dans un fichier `updateinfo.xml` (ou un package contenant un fichier sans valeurs Classification, Severity et Date correctement formatées) peuvent être inclus dans la liste des correctifs préfiltrée. Toutefois, pour qu'un correctif soit appliqué, il doit toujours satisfaire aux règles de référence de correctif spécifiées par l'utilisateur.

CentOS

Sur CentOS, le service de référence de correctif Systems Manager utilise des référentiels (repos) préconfigurés sur l'instance. Voici quelques exemples d'Amazon Machine Image (AMI) CentOS 6.9 :

- ID du référentiel : ultra-centos-6.9-base

  Nom du référentiel : UltraServeCentOS-6.9 - Base
- ID du référentiel : ultra-centos-6.9-extras

  Nom du référentiel : UltraServeCentOS-6.9 - Extras
- ID du référentiel : ultra-centos-6.9-updates

  Nom du référentiel : UltraServeCentOS-6.9 - Mises à jour
- ID du référentiel : ultra-centos-6.x-glusterfs

  Nom du référentiel : UltraServeCentOS-6.x -GlusterFS
- ID du référentiel : ultra-centos-6.x-ultrarepo

  Nom du référentiel : UltraServeCentOS-6.x – Packages UltraServe de référentiel

> **Note**
>
> Toutes les mises à jour sont téléchargées à partir des référentiels distants configurés sur l'instance. Par conséquent, l'instance doit être en mesure de se connecter aux référentiels afin que l'application de correctifs s'exécute.

CentOSLes instances 6 et 7 utilisent Yum comme gestionnaire de package. CentOS 8 instances utilisent DNF comme gestionnaire de packages. Les deux gestionnaires de packages utilisent le concept d'une notice de mise à jour. Une notice de mise à jour est simplement un ensemble de packages qui corrigent des problèmes spécifiques. Tous les packages qui sont inclus dans une notice de mise à jour sont considérés comme des packages de sécurité par Gestionnaire de correctifs .

Cependant, les référentiels CentOS par défaut ne sont pas configurés avec une notice de mise à jour. Cela signifie que Gestionnaire de correctifs ne détecte pas les packages sur un CentOS référentiel par

défaut. Pour permettre Gestionnaire de correctifs à de traiter les packages qui ne sont pas contenus dans une notice de mise à jour, vous devez activer l'`EnableNonSecurity`indicateur dans les règles de référence de correctif.

> **Note**
>
> Les notices de CentOS mise à jour d' sont prises en charge. Les référentiels avec des notices de mise à jour peuvent être téléchargés après le lancement.

Debian Server

Sur Debian Server, le service de référence de correctif Systems Manager utilise des référentiels (repos) préconfigurés sur l'instance. Ces référentiels préconfigurés sont utilisés pour extraire une liste mise à jour des mises à niveau de package disponibles. Pour cela, Systems Manager exécute l'équivalent d'une commande `sudo apt-get update`.

Les packages sont ensuite filtrés à partir des `debian-security` *`codename`* référentiels, où le nom de code est similaire à ou `jessie` .`stretch` Par exemple, sur Debian Server 8, identifie Gestionnaire de correctifs uniquement les mises à niveau faisant partie d' `debian-security jessie` . Sur Debian Server 9, seules les mises à niveau faisant partie de `debian-security stretch` sont identifiées.

> **Note**
>
> Sur Debian Server 8 uniquement : Étant donné que certaines instances Debian Server 8.* font référence à un référentiel de packages obsolète (`jessie-backports`), Gestionnaire de correctifs effectue des étapes supplémentaires pour s'assurer que les opérations d'application de correctifs réussissent. Pour plus d'informations, consultez la section Installation des correctifs (p. 1056).

Oracle Linux

Sur Oracle Linux, le service de référence de correctif Systems Manager utilise des référentiels (repos) préconfigurés sur l'instance. Une instance comporte généralement deux référentiels préconfigurés :

- ID de référentiel : OL7_UEKR5/x86_64

  Nom du référentiel : Dernière version Unbreakable Enterprise Kernel Version 5 pour Oracle Linux 7Server (x86_64)
- ID du référentiel : ol7_latest/x86_64

  Nom du référentiel :Oracle Linux 7Dernière version du serveur (x86_64)

> **Note**
>
> Toutes les mises à jour sont téléchargées à partir des référentiels distants configurés sur l'instance. Par conséquent, l'instance doit être en mesure de se connecter aux référentiels afin que l'application de correctifs s'exécute.

Oracle LinuxLes instances utilisent Yum comme gestionnaire de package et Yum utilise le concept de notice de mise à jour sous la forme d'un fichier nommé `updateinfo.xml` . Une notice de mise à jour est simplement un ensemble de packages qui corrigent des problèmes spécifiques. Tous les packages qui sont inclus dans une notice de mise à jour sont considérés comme des packages de sécurité par Gestionnaire de correctifs . Les packages ne se voient pas attribuer des classifications ou des niveaux de gravité. Pour cette raison, Gestionnaire de correctifs attribue les attributs d'une notice de mise à jour aux packages associés.

> **Note**
>
> Si vous cochez la case Approved patches include non-security updates (Les correctifs approuvés incluent des mises à jour non liées à la sécurité) dans la page Create patch

baseline (Créer une référence de correctif), les packages qui ne sont pas classées dans un fichier `updateinfo.xml` (ou un package contenant un fichier sans valeurs Classification, Severity et Date correctement formatées) peuvent être inclus dans la liste des correctifs préfiltrée. Toutefois, pour qu'un correctif soit appliqué, il doit toujours satisfaire aux règles de référence de correctif spécifiées par l'utilisateur.

RHEL

Sur Red Hat Enterprise Linux, le service de référence de correctif Systems Manager utilise des référentiels (repos) préconfigurés sur l'instance. Une instance comporte généralement trois référentiels préconfigurés.

Toutes les mises à jour sont téléchargées à partir des référentiels distants configurés sur l'instance. Par conséquent, l'instance doit être en mesure de se connecter aux référentiels afin que l'application de correctifs s'exécute.

> **Note**
>
> Si vous cochez la case Approved patches include non-security updates (Les correctifs approuvés incluent des mises à jour non liées à la sécurité) dans la page Create patch baseline (Créer une référence de correctif), les packages qui ne sont pas classées dans un fichier `updateinfo.xml` (ou un package contenant un fichier sans valeurs Classification, Severity et Date correctement formatées) peuvent être inclus dans la liste des correctifs préfiltrée. Toutefois, pour qu'un correctif soit appliqué, il doit toujours satisfaire aux règles de référence de correctif spécifiées par l'utilisateur.

Red Hat Enterprise Linux7 instances utilisent Yum comme gestionnaire de package. Red Hat Enterprise Linux 8 instances utilisent DNF comme gestionnaire de package. Les deux gestionnaires de packages utilisent le concept d'un avis de mise à jour sous la forme d'un fichier nommé `updateinfo.xml` . Une notice de mise à jour est simplement un ensemble de packages qui corrigent des problèmes spécifiques. Tous les packages qui sont inclus dans une notice de mise à jour sont considérés comme des packages de sécurité par Gestionnaire de correctifs . Les packages ne se voient pas attribuer des classifications ou des niveaux de gravité. Pour cette raison, Gestionnaire de correctifs attribue les attributs d'une notice de mise à jour aux packages associés.

Notez que les emplacements de référentiel diffèrent entre RHEL 7 et RHEL 8 :

RHEL 7

> **Note**
>
> Le référentiel suivant IDs est associé à RHUI 2. RHUI 3 a été lancé en décembre 2019 et a introduit un autre schéma d'attribution de noms pour le référentiel YumIDs. Selon l'AMI RHEL-7 à partir de laquelle vous créez vos instances, vous devrez peut-être mettre à jour vos commandes. Pour plus d'informations, consultez Référentiel IDs pour RHEL 7 dans AWS Have Changed sur le portail client Red Hat.

- ID du référentiel : rhui-REGION-client-config-server-7/x86_64

  Nom du référentiel : Red Hat Update Infrastructure 2.0 Client Configuration Server 7
- ID du référentiel : rhui-REGION-rhel-server-releases/7Server/x86_64

  Nom du référentiel : Red Hat Enterprise Linux Server 7 (RPM)
- ID du référentiel : rhui-REGION-rhel-server-rh-common/7Server/x86_64

  Nom du référentiel : Propriété Common (RPM) de Red Hat Enterprise Linux Server 7 RH

RHEL 8

- ID du référentiel : rhel-8-appstream-rhui-rpms

  Nom du référentiel : Red Hat Enterprise Linux 8 pour x86_64 - AppStream de RHUI (RPM)

- ID du référentiel : rhel-8-baseos-rhui-rpms

  Nom du référentiel : Red Hat Enterprise Linux 8 pour x86_64 - BaseOS de RHUI (RPM)
- ID du référentiel : rhui-client-config-server-8

  Nom du référentiel : Red Hat Update Infrastructure 3 Client Configuration Server 8

SLES

Sur les instances SUSE Linux Enterprise Server (SLES), la bibliothèque ZYPP obtient la liste des correctifs disponibles (un ensemble de packages) à partir des emplacements suivants :

- Liste de référentiels : `etc/zypp/repos.d/*`
- Informations sur les packages : `/var/cache/zypp/raw/*`

Les instances SLES utilisent Zypper comme gestionnaire de package et Zypper utilise le concept de correctif. Un correctif est simplement un ensemble de packages qui corrigent un problème spécifique. Gestionnaire de correctifs gère tous les packages référencés dans un correctif comme liés à la sécurité. Étant donné que les packages individuels ne sont associés à aucune classification ou gravité, Gestionnaire de correctifs attribue aux packages les attributs du correctif auquel ils appartiennent.

Ubuntu Server

Sur Ubuntu Server, le service de référence de correctif Systems Manager utilise des référentiels (repos) préconfigurés sur l'instance. Ces référentiels préconfigurés sont utilisés pour extraire une liste mise à jour des mises à niveau de package disponibles. Pour cela, Systems Manager exécute l'équivalent d'une commande `sudo apt-get update`.

Les packages sont ensuite filtrés à partir des *`codename`*`-security` référentiels, où le nom de code est unique à la version, par exemple `trusty` pour Ubuntu Server 14. Identifie Gestionnaire de correctifs uniquement les mises à niveau qui font partie de ces référentiels :

- Ubuntu Server14 :`trusty-security`
- Ubuntu Server16 :`xenial-security`
- Ubuntu Server18 :`bionic-security`
- Ubuntu Server20 :`focal-security`

Windows

Sur les systèmes d'exploitation Microsoft Windows, Gestionnaire de correctifs extrait une liste des mises à jour disponibles que Microsoft publie dans Microsoft Update et est automatiquement disponible pour Windows Server Update Services (WSUS).

surveille Gestionnaire de correctifs en permanence les nouvelles mises à jour dans chaque région AWS. La liste des mises à jour disponibles est actualisée dans chaque région au moins une fois par jour. Lorsque les informations de correctif de Microsoft sont traitées, Gestionnaire de correctifs supprime de la liste des correctifs les mises à jour qui ont été remplacées par des mises à jour ultérieures. Par conséquent, seule la mise à jour la plus récente est affichée et disponible pour être installée. Par exemple, si `KB4012214` remplace `KB3135456` , seule est disponible comme mise à jour dans `KB4012214` .Gestionnaire de correctifs

> **Note**
>
> rend Gestionnaire de correctifs uniquement disponible les correctifs pour les versions du système Windows Server d'exploitation prises en charge pour Gestionnaire de correctifs . Par exemple, ne Gestionnaire de correctifs peut pas être utilisé pour corriger Windows RT.

Spécification d'un autre référentiel source de correctifs (Linux)

Lorsque vous utilisez les référentiels par défaut configurés sur une instance pour les opérations d'application de correctifs, Gestionnaire de correctifs analyse ou installe les correctifs liés à la sécurité. Il s'agit du comportement par défaut pour Gestionnaire de correctifs . Pour plus d'informations sur la façon dont Gestionnaire de correctifs sélectionne et installe les correctifs de sécurité, consultez Sélection des correctifs de sécurité (p. 1049) .

Sur les systèmes Linux, toutefois, vous pouvez également utiliser Gestionnaire de correctifs pour installer des correctifs qui ne sont pas liés à la sécurité, ou qui sont dans un référentiel source différent de celui par défaut configuré sur l'instance. Vous pouvez spécifier d'autres référentiels source de correctifs lors de la création d'une référence de correctifs personnalisée. Dans chaque référence de correctifs personnalisée, vous pouvez spécifier des configurations source de correctifs pour un maximum de 20 versions d'un système d'exploitation Linux pris en charge.

Par exemple, supposons que votre parc Ubuntu Server contienne à la fois les instances Ubuntu Server 16.04 et Ubuntu Server 14.04. Dans ce cas, vous pouvez spécifier d'autres référentiels pour chaque version dans la même référence de correctifs personnalisée. Pour chaque version, vous fournissez un nom, spécifiez le type de version du système d'exploitation (produit) et indiquez une configuration de référentiel. Vous pouvez également spécifier un autre référentiel source unique, qui s'applique à toutes les versions d'un système d'exploitation pris en charge.

> **Note**
>
> L'exécution d'une référence de correctifs personnalisée qui spécifie d'autres référentiels de correctifs sur une instance ne modifie pas le référentiel par défaut configuré pour l'instance.

Pour obtenir la liste des exemples de scénarios pour l'utilisation de cette option, consultez Exemples d'utilisations pour d'autres référentiels source de correctifs (p. 1055) plus loin dans cette rubrique.

Pour plus d'informations sur les références de correctifs par défaut et personnalisées, consultez À propos des références de correctifs prédéfinies et personnalisées (p. 1098).

Exemple: Utilisation de la console

Pour spécifier d'autres référentiels source de correctifs lorsque vous travaillez dans la console AWS Systems Manager, utilisez la section Patch sources (Sources des correctifs) de la page Create patch baseline (Créer une référence de correctif). Pour de plus amples informations sur l'utilisation des options Sources des correctifs, veuillez consulter Création d'une référence de correctifs personnalisée (Linux) (p. 1130).

Exemple: Utilisation du AWS CLI

Pour un exemple d'utilisation de l'option `--sources` à l'aide de l'interface de ligne de commande, consultez Création d'une référence de correctifs avec des référentiels personnalisés pour les différentes versions du système d'exploitation (p. 1149).

Rubriques

- Points importants à prendre en compte pour d'autres référentiels (p. 1054)
- Exemples d'utilisations pour d'autres référentiels source de correctifs (p. 1055)

Points importants à prendre en compte pour d'autres référentiels

Gardez les points suivants à l'esprit lorsque vous planifiez votre stratégie d'application de correctifs en utilisant d'autres référentiels de correctifs.

Seuls les référentiels spécifiés sont utilisés pour l'application de correctifs

Spécifier d'autres référentiels ne signifie pas spécifier des référentiels supplémentaires. Vous pouvez choisir de spécifier des référentiels autres que ceux configurés par défaut sur une instance. Toutefois, vous

devez également spécifier les référentiels par défaut dans le cadre de la configuration d'autres sources de correctifs si vous voulez que leurs mises à jour soient appliquées.

Par exemple, sur Amazon Linux 2 les instances , les référentiels par défaut sont `amzn-main` et `amzn-update` . Si vous souhaitez inclure le référentiel EPEL (Extra Packages for Enterprise Linux) dans vos opérations d'application de correctifs, vous devez spécifier les trois référentiels en tant que référentiels alternatifs.

> **Note**
>
> L'exécution d'une référence de correctifs personnalisée qui spécifie d'autres référentiels de correctifs sur une instance ne modifie pas le référentiel par défaut configuré pour l'instance.

Le comportement d'application de correctifs pour les distributions basées sur YUM dépend du manifeste updateinfo.xml

Lorsque vous spécifiez d'autres référentiels de correctifs pour les distributions basées sur YUM, comme Amazon Linux ou Amazon Linux 2 , ou Red Hat Enterprise Linux , le comportement d'application des correctifs varie selon que le référentiel inclut ou non un manifeste de mise à jour sous la forme d'un CentOS fichier complet et correctement formaté.`updateinfo.xml` Ce fichier spécifie la date de parution, la classification et la gravité des différents packages. Les éléments suivants ont un impact sur le comportement d'application des correctifs :

- Si vous filtrez sur la classification et la gravité, mais qu'elles ne sont pas spécifiées dans `updateinfo.xml`, le package ne sera pas inclus par le filtre. Cela signifie également que les packages sans fichier `updateinfo.xml` ne seront pas inclus dans l'application des correctifs.
- Si vous filtrez sur ApprovalAfterDays , mais que la date de parution du package n'est pas au format Unix Epoch (ou n'a pas de date de parution spécifiée), le package n'est pas inclus par le filtre.
- Il existe une exception si vous cochez la case Approved patches include non-security updates (Les correctifs approuvés incluent des mises à jour non liées à la sécurité) dans la page Create patch baseline (Créer une référence de correctif). Dans ce cas, les packages sans fichier `updateinfo.xml` (ou qui contiennent ce fichier sans valeurs correctement formatées de Classification, Severity et Date) seront inclus dans la liste préfiltrée des correctifs. (Ils doivent encore satisfaire aux autres conditions préalables relatives aux règles de référence de correctifs pour pouvoir être installés.)

Exemples d'utilisations pour d'autres référentiels source de correctifs

Exemple 1 - Mises à jour non liées à la sécurité pour Ubuntu Server

Vous utilisez déjà Gestionnaire de correctifs pour installer des correctifs de sécurité sur un parc d'Ubuntu Serverinstances à l'aide de la référence de correctifs prédéfinie fournie par AWS : AWS-UbuntuDefaultPatchBaseline. Vous pouvez créer une référence de correctifs basée sur cette valeur par défaut, mais spécifiez dans les règles d'approbation que vous ne voulez pas que les mises à jour non liées à la sécurité qui font partie de la distribution par défaut soient également installées. Lorsque cette référence de correctifs est exécutée sur vos instances, les correctifs liés à la sécurité et non liés à la sécurité sont tous appliqués. Vous pouvez également choisir d'approuver les correctifs non liés à la sécurité dans les exceptions de correctif que vous spécifiez pour une référence.

Exemple 2 - Dépôts PPA (Personal Package Archive) pour Ubuntu Server

Vos instances Ubuntu Server exécutent des logiciels qui sont distribués par le biais de dépôts Personal Package Archive (PPA) pour Ubuntu. Dans ce cas, vous créez une référence de correctifs qui spécifie un référentiel PPA que vous avez configuré sur l'instance en tant que référentiel source pour l'application des correctifs. Utilisez ensuite la fonctionnalité Fonctionnalité Exécuter la commande pour exécuter le document de référence de correctifs sur les instances.

Exemple 3 - Applications d'entreprise internes sur Amazon Linux

Vous devez exécuter certaines applications nécessaires pour être en conformité avec les réglementations du secteur sur vos instances Amazon Linux. Vous pouvez configurer un référentiel pour ces applications

sur les instances, utiliser YUM pour installer en premier lieu les applications, puis mettre à jour ou créer une référence de correctifs pour inclure ce nouveau référentiel d'entreprise. Après cela, vous pouvez utiliser la fonctionnalité Fonctionnalité Exécuter la commande pour exécuter le document AWS-RunPatchBaseline avec l'option `Scan` pour voir si le package d'entreprise est répertorié parmi les packages installés et à jour sur l'instance. S'il n'est pas à jour, vous pouvez exécuter à nouveau le document à l'aide de l'option `Install` pour mettre à jour les applications.

Installation des correctifs

Gestionnaire de correctifsutilise le mécanisme intégré approprié pour un type de système d'exploitation pour installer les mises à jour sur une instance. Par exemple, sous Windows Server, l'API Windows Update est utilisée, et sous Amazon Linux, le gestionnaire de package `yum` est utilisé.

Le reste de cette section explique comment Gestionnaire de correctifs installe les correctifs sur un système d'exploitation.

Amazon Linux and Amazon Linux 2

Sur les instances Amazon Linux et Amazon Linux 2, le flux d'installation de correctifs est le suivant :

1. Appliquez des filtres GlobalFilters comme indiqué dans la référence de correctif en conservant uniquement les packages qualifiés à des fins de traitement ultérieur.
2. Appliquez ApprovalRules comme indiqué dans la référence de correctif. Chaque règle d'approbation peut définir un package comme approuvé.

   Toutefois, les règles d'approbation varient également selon que la case à cocher Inclusion de mises à jour non liées à la sécurité a été ou non activée lors de la création ou de la dernière mise à jour d'une référence de correctif.

   Si les mises à jour non liées à la sécurité sont exclues, une règle implicite est appliquée afin de sélectionner uniquement les packages avec des mises à niveau dans les référentiels de sécurité. Pour chaque package, la version du package proposée (généralement la plus récente) doit se trouver dans un référentiel de sécurité.

   Si les mises à jour non liées à la sécurité sont incluses, les correctifs provenant d'autres référentiels sont également pris en compte.
3. Appliquez Correctifs approuvés comme indiqué dans la référence de correctif. Les correctifs approuvés le sont pour une mise à jour, même s'ils sont ignorés par GlobalFilters ou si aucune règle d'approbation spécifiée dans ApprovalRules ne leur accorde d'approbation.
4. Appliquez RejectedPatches comme indiqué dans la référence de correctif. Les correctifs rejetés sont supprimés de la liste des correctifs approuvés et ne seront pas appliqués.
5. Si plusieurs versions d'un correctif sont approuvées, la version la plus récente sera appliquée.
6. L'API de mise à jour YUM est appliquée aux correctifs approuvés comme suit :
   - Pour les références de correctifs par défaut prédéfinies fournies par AWS et pour les références de correctifs personnalisées pour lesquelles la case Approved patches include non-security updates (les correctifs approuvés incluent des mises à jour non liées à la sécurité) n'est pas cochée, seuls les correctifs spécifiés dans `updateinfo.xml` sont appliqués (mises à jour de sécurité uniquement).

     La commande yum équivalente pour ce flux de travail est :

     ```
     sudo yum update-minimal --sec-severity=critical,important --bugfix -y
     ```
   - Pour les références de correctifs personnalisés pour lesquelles l'option Approved patches include non-security updates (Les correctifs approuvés incluent des mises à jour non liées à la sécurité) est sélectionnée, les correctifs figurant dans `updateinfo.xml` et ceux ne figurant pas dans `updateinfo.xml` sont appliqués (mises à jour de sécurité et non liées à la sécurité).

La commande yum équivalente pour ce flux de travail est :

```
sudo yum update --security --bugfix
```

7. L'instance est redémarrée si des mises à jour ont été installées. (Exception : si le paramètre `RebootOption` est défini sur `NoReboot` dans le document `AWS-RunPatchBaseline`, l'instance n'est pas redémarrée après l'exécution du Gestionnaire de correctifs. Pour de plus amples informations, veuillez consulter Nom du paramètre : `RebootOption` (p. 1087).)

CentOS

Sur les instances CentOS, le flux d'installation de correctifs est le suivant :

1. Appliquez des filtres GlobalFilters comme indiqué dans la référence de correctif en conservant uniquement les packages qualifiés à des fins de traitement ultérieur.
2. Appliquez ApprovalRules comme indiqué dans la référence de correctif. Chaque règle d'approbation peut définir un package comme approuvé.

   Toutefois, les règles d'approbation varient également selon que la case à cocher Inclusion de mises à jour non liées à la sécurité a été ou non activée lors de la création ou de la dernière mise à jour d'une référence de correctif.

   Si les mises à jour non liées à la sécurité sont exclues, une règle implicite est appliquée afin de sélectionner uniquement les packages avec des mises à niveau dans les référentiels de sécurité. Pour chaque package, la version du package proposée (généralement la plus récente) doit se trouver dans un référentiel de sécurité.

   Si les mises à jour non liées à la sécurité sont incluses, les correctifs provenant d'autres référentiels sont également pris en compte.
3. Appliquez Correctifs approuvés comme indiqué dans la référence de correctif. Les correctifs approuvés le sont pour une mise à jour, même s'ils sont ignorés par GlobalFilters ou si aucune règle d'approbation spécifiée dans ApprovalRules ne leur accorde d'approbation.
4. Appliquez RejectedPatches comme indiqué dans la référence de correctif. Les correctifs rejetés sont supprimés de la liste des correctifs approuvés et ne seront pas appliqués.
5. Si plusieurs versions d'un correctif sont approuvées, la version la plus récente sera appliquée.
6. L'API de mise à jour YUM (sur les versions CentOS 6.x et 7.x) ou la mise à jour DNF (sur CentOS 8) est appliquée aux correctifs approuvés.
7. L'instance est redémarrée si des mises à jour ont été installées. (Exception : si le paramètre `RebootOption` est défini sur `NoReboot` dans le document `AWS-RunPatchBaseline`, l'instance n'est pas redémarrée après l'exécution du Gestionnaire de correctifs. Pour de plus amples informations, veuillez consulter Nom du paramètre : `RebootOption` (p. 1087).)

Debian Server

Sur les instances Debian Server, le flux d'installation de correctifs est le suivant :

1. Sur Debian Server 8 uniquement : Étant donné que certaines instances Debian Server 8.* font référence à un référentiel de packages obsolète (`jessie-backports`), Gestionnaire de correctifs effectue les étapes supplémentaires suivantes pour s'assurer que les opérations d'application de correctifs réussissent :
   a. Sur votre instance, la référence au référentiel `jessie-backports` est commentée à partir de la liste des emplacements source (`/etc/apt/sources.list.d/jessie-backports`). Par conséquent, aucune tentative n'est effectuée pour télécharger des correctifs à partir de cet emplacement.

b. Une clé de signature de mise à jour de sécurité Stretch est importée. Cette clé fournit les autorisations nécessaires pour les opérations de mise à jour et d'installation sur les distributions Debian Server 8.*.

c. Une opération `apt-get` est exécutée pour s'assurer que la dernière version de `python3-apt` est installée avant le début du processus d'application des correctifs.

d. Une fois le processus d'installation terminé, la référence au référentiel `jessie-backports` est restaurée et la clé de signature est supprimée du porte-clés source apt. Ainsi, la configuration du système demeure telle qu'elle était avant l'opération d'application de correctifs.

La prochaine fois que Gestionnaire de correctifs met à jour le système, le même processus est répété.

Les étapes restantes du flux d'installation de correctifs s'appliquent aux versions Debian Server 8 et 9.

2. Appliquez des filtres GlobalFilters comme indiqué dans la référence de correctif en conservant uniquement les packages qualifiés à des fins de traitement ultérieur.
3. Appliquez ApprovalRules comme indiqué dans la référence de correctif. Chaque règle d'approbation peut définir un package comme approuvé.

   Note

   Comme il n'est pas possible de déterminer de manière fiable les dates de publication des packages de mise à jour pour Debian Server, les options d'approbation automatique ne sont pas prises en charge pour ce système d'exploitation.

   Toutefois, les règles d'approbation varient également selon que la case à cocher Inclusion de mises à jour non liées à la sécurité a été ou non activée lors de la création ou de la dernière mise à jour d'une référence de correctif.

   Si les mises à jour non liées à la sécurité sont exclues, une règle implicite est appliquée afin de sélectionner uniquement les packages avec des mises à niveau dans les référentiels de sécurité. Pour chaque package, la version du package proposée (généralement la plus récente) doit se trouver dans un référentiel de sécurité.

   Si les mises à jour non liées à la sécurité sont incluses, les correctifs provenant d'autres référentiels sont également pris en compte.

   Note

   Pour Debian Server , les versions de correctifs candidats sont limitées aux correctifs inclus dans `debian-security` .
4. Appliquez Correctifs approuvés comme indiqué dans la référence de correctif. Les correctifs approuvés le sont pour une mise à jour, même s'ils sont ignorés par GlobalFilters ou si aucune règle d'approbation spécifiée dans ApprovalRules ne leur accorde d'approbation.
5. Appliquez RejectedPatches comme indiqué dans la référence de correctif. Les correctifs rejetés sont supprimés de la liste des correctifs approuvés et ne seront pas appliqués.
6. La bibliothèque APT permet de mettre à niveau les packages.
7. L'instance est redémarrée si des mises à jour ont été installées. (Exception : si le paramètre `RebootOption` est défini sur `NoReboot` dans le document `AWS-RunPatchBaseline`, l'instance n'est pas redémarrée après l'exécution du Gestionnaire de correctifs. Pour de plus amples informations, veuillez consulter Nom du paramètre : `RebootOption` (p. 1087).)

Oracle Linux

Sur les instances Oracle Linux, le flux d'installation de correctifs est le suivant :

1. Appliquez des filtres GlobalFilters comme indiqué dans la référence de correctif en conservant uniquement les packages qualifiés à des fins de traitement ultérieur.
2. Appliquez ApprovalRules comme indiqué dans la référence de correctif. Chaque règle d'approbation peut définir un package comme approuvé.

   Toutefois, les règles d'approbation varient également selon que la case à cocher Inclusion de mises à jour non liées à la sécurité a été ou non activée lors de la création ou de la dernière mise à jour d'une référence de correctif.

   Si les mises à jour non liées à la sécurité sont exclues, une règle implicite est appliquée afin de sélectionner uniquement les packages avec des mises à niveau dans les référentiels de sécurité. Pour chaque package, la version du package proposée (généralement la plus récente) doit se trouver dans un référentiel de sécurité.

   Si les mises à jour non liées à la sécurité sont incluses, les correctifs provenant d'autres référentiels sont également pris en compte.
3. Appliquez Correctifs approuvés comme indiqué dans la référence de correctif. Les correctifs approuvés le sont pour une mise à jour, même s'ils sont ignorés par GlobalFilters ou si aucune règle d'approbation spécifiée dans ApprovalRules ne leur accorde d'approbation.
4. Appliquez RejectedPatches comme indiqué dans la référence de correctif. Les correctifs rejetés sont supprimés de la liste des correctifs approuvés et ne seront pas appliqués.
5. Si plusieurs versions d'un correctif sont approuvées, la version la plus récente sera appliquée.
6. L'API de mise à jour YUM est appliquée aux correctifs approuvés comme suit :
   - Pour les références de correctifs par défaut prédéfinies fournies par AWS et pour les références de correctifs personnalisées pour lesquelles la case Approved patches include non-security updates (les correctifs approuvés incluent des mises à jour non liées à la sécurité) n'est pas cochée, seuls les correctifs spécifiés dans `updateinfo.xml` sont appliqués (mises à jour de sécurité uniquement).

     La commande yum équivalente pour ce flux de travail est :

     ```
     sudo yum update-minimal --sec-severity=important,moderate --bugfix -y
     ```
   - Pour les références de correctifs personnalisés pour lesquelles l'option Approved patches include non-security updates (Les correctifs approuvés incluent des mises à jour non liées à la sécurité) est sélectionnée, les correctifs figurant dans `updateinfo.xml` et ceux ne figurant pas dans `updateinfo.xml` sont appliqués (mises à jour de sécurité et non liées à la sécurité).

     La commande yum équivalente pour ce flux de travail est :

     ```
     sudo yum update --security --bugfix -y
     ```
7. L'instance est redémarrée si des mises à jour ont été installées. (Exception : si le paramètre `RebootOption` est défini sur `NoReboot` dans le document `AWS-RunPatchBaseline`, l'instance n'est pas redémarrée après l'exécution du Gestionnaire de correctifs. Pour de plus amples informations, veuillez consulter Nom du paramètre : `RebootOption` (p. 1087).)

RHEL

Sur les instances Red Hat Enterprise Linux, le flux d'installation de correctifs est le suivant :

1. Appliquez des filtres GlobalFilters comme indiqué dans la référence de correctif en conservant uniquement les packages qualifiés à des fins de traitement ultérieur.
2. Appliquez ApprovalRules comme indiqué dans la référence de correctif. Chaque règle d'approbation peut définir un package comme approuvé.

Toutefois, les règles d'approbation varient également selon que la case à cocher Inclusion de mises à jour non liées à la sécurité a été ou non activée lors de la création ou de la dernière mise à jour d'une référence de correctif.

Si les mises à jour non liées à la sécurité sont exclues, une règle implicite est appliquée afin de sélectionner uniquement les packages avec des mises à niveau dans les référentiels de sécurité. Pour chaque package, la version du package proposée (généralement la plus récente) doit se trouver dans un référentiel de sécurité.

Si les mises à jour non liées à la sécurité sont incluses, les correctifs provenant d'autres référentiels sont également pris en compte.

3. Appliquez Correctifs approuvés comme indiqué dans la référence de correctif. Les correctifs approuvés le sont pour une mise à jour, même s'ils sont ignorés par GlobalFilters ou si aucune règle d'approbation spécifiée dans ApprovalRules ne leur accorde d'approbation.
4. Appliquez RejectedPatches comme indiqué dans la référence de correctif. Les correctifs rejetés sont supprimés de la liste des correctifs approuvés et ne seront pas appliqués.
5. Si plusieurs versions d'un correctif sont approuvées, la version la plus récente sera appliquée.
6. L'API de mise à jour YUM (sur RHEL 7) ou l'API de mise à jour DNF (sur RHEL 8) est appliquée aux correctifs approuvés comme suit :
   - Pour les références de correctifs par défaut prédéfinies fournies par AWS et pour les références de correctifs personnalisées pour lesquelles la case Approved patches include non-security updates (les correctifs approuvés incluent des mises à jour non liées à la sécurité) n'est pas cochée, seuls les correctifs spécifiés dans `updateinfo.xml` sont appliqués (mises à jour de sécurité uniquement).

     Pour RHEL 7, la commande yum équivalente pour ce flux de travail est :

     ```
     sudo yum update-minimal --sec-severity=critical,important --bugfix -y
     ```

     Pour RHEL 8, les commandes dnf équivalentes pour ce flux de travail sont les suivantes :

     ```
     sudo dnf update-minimal --sec-severity=Critical --bugfix -y ; \
     sudo dnf update-minimal --sec-severity=Important --bugfix -y
     ```

   - Pour les références de correctifs personnalisés pour lesquelles l'option Approved patches include non-security updates (Les correctifs approuvés incluent des mises à jour non liées à la sécurité) est sélectionnée, les correctifs figurant dans `updateinfo.xml` et ceux ne figurant pas dans `updateinfo.xml` sont appliqués (mises à jour de sécurité et non liées à la sécurité).

     Pour RHEL 7, la commande yum équivalente pour ce flux de travail est :

     ```
     sudo yum update --security --bugfix -y
     ```

     Pour RHEL 8, la commande yum équivalente pour ce flux de travail est :

     ```
     sudo dnf update --security --bugfix -y
     ```

7. L'instance est redémarrée si des mises à jour ont été installées. (Exception : si le paramètre `RebootOption` est défini sur `NoReboot` dans le document `AWS-RunPatchBaseline`, l'instance n'est pas redémarrée après l'exécution du Gestionnaire de correctifs. Pour de plus amples informations, veuillez consulter Nom du paramètre : `RebootOption` (p. 1087).)

SLES

Sur les instances SUSE Linux Enterprise Server (SLES), le flux d'installation de correctifs est le suivant :

1. Appliquez des filtres GlobalFilters comme indiqué dans la référence de correctif en conservant uniquement les packages qualifiés à des fins de traitement ultérieur.
2. Appliquez ApprovalRules comme indiqué dans la référence de correctif. Chaque règle d'approbation peut définir un package comme approuvé.

   Toutefois, les règles d'approbation varient également selon que la case à cocher Inclusion de mises à jour non liées à la sécurité a été ou non activée lors de la création ou de la dernière mise à jour d'une référence de correctif.

   Si les mises à jour non liées à la sécurité sont exclues, une règle implicite est appliquée afin de sélectionner uniquement les packages avec des mises à niveau dans les référentiels de sécurité. Pour chaque package, la version du package proposée (généralement la plus récente) doit se trouver dans un référentiel de sécurité.

   Si les mises à jour non liées à la sécurité sont incluses, les correctifs provenant d'autres référentiels sont également pris en compte.
3. Appliquez Correctifs approuvés comme indiqué dans la référence de correctif. Les correctifs approuvés le sont pour une mise à jour, même s'ils sont ignorés par GlobalFilters ou si aucune règle d'approbation spécifiée dans ApprovalRules ne leur accorde d'approbation.
4. Appliquez RejectedPatches comme indiqué dans la référence de correctif. Les correctifs rejetés sont supprimés de la liste des correctifs approuvés et ne seront pas appliqués.
5. Si plusieurs versions d'un correctif sont approuvées, la version la plus récente sera appliquée.
6. L'API de mise à jour Zypper est appliquée aux correctifs approuvés.
7. L'instance est redémarrée si des mises à jour ont été installées. (Exception : si le paramètre `RebootOption` est défini sur `NoReboot` dans le document `AWS-RunPatchBaseline`, l'instance n'est pas redémarrée après l'exécution du Gestionnaire de correctifs. Pour de plus amples informations, veuillez consulter Nom du paramètre : `RebootOption` (p. 1087).)

Ubuntu Server

Sur les instances Ubuntu Server, le flux d'installation de correctifs est le suivant :

1. Appliquez des filtres GlobalFilters comme indiqué dans la référence de correctif en conservant uniquement les packages qualifiés à des fins de traitement ultérieur.
2. Appliquez ApprovalRules comme indiqué dans la référence de correctif. Chaque règle d'approbation peut définir un package comme approuvé.

   > **Note**
   >
   > Comme il n'est pas possible de déterminer de manière fiable les dates de publication des packages de mise à jour pour Ubuntu Server, les options d'approbation automatique ne sont pas prises en charge pour ce système d'exploitation.

   Toutefois, les règles d'approbation varient également selon que la case à cocher Inclusion de mises à jour non liées à la sécurité a été ou non activée lors de la création ou de la dernière mise à jour d'une référence de correctif.

   Si les mises à jour non liées à la sécurité sont exclues, une règle implicite est appliquée afin de sélectionner uniquement les packages avec des mises à niveau dans les référentiels de sécurité. Pour chaque package, la version du package proposée (généralement la plus récente) doit se trouver dans un référentiel de sécurité.

Si les mises à jour non liées à la sécurité sont incluses, les correctifs provenant d'autres référentiels sont également pris en compte.

Toutefois, les règles d'approbation sont également soumises à l'activation ou à la dernière case à cocher d'inclusion des mises à jour non liées à la sécurité lors de la création ou de la dernière mise à jour d'une référence de correctifs.

> **Note**
>
> Pour Ubuntu Server , les versions des correctifs candidats sont limitées aux correctifs inclus dans `trusty-security` (Ubuntu Server 14), `xenial-security` (Ubuntu Server 16), `bionic-security` (Ubuntu Server 18) ou `focal-security` (Ubuntu Server 20).

3. Appliquez Correctifs approuvés comme indiqué dans la référence de correctif. Les correctifs approuvés le sont pour une mise à jour, même s'ils sont ignorés par GlobalFilters ou si aucune règle d'approbation spécifiée dans ApprovalRules ne leur accorde d'approbation.
4. Appliquez RejectedPatches comme indiqué dans la référence de correctif. Les correctifs rejetés sont supprimés de la liste des correctifs approuvés et ne seront pas appliqués.
5. La bibliothèque APT permet de mettre à niveau les packages.
6. L'instance est redémarrée si des mises à jour ont été installées. (Exception : si le paramètre `RebootOption` est défini sur `NoReboot` dans le document `AWS-RunPatchBaseline`, l'instance n'est pas redémarrée après l'exécution du Gestionnaire de correctifs. Pour de plus amples informations, veuillez consulter Nom du paramètre : `RebootOption` (p. 1087).)

Windows

Lors de l'exécution d'une opération d'application de correctifs sur une instance Windows Server, l'instance demande un instantané de la référence de correctif correspondante dans Systems Manager. Cet instantané contient la liste de toutes les mises à jour disponibles dans la référence de correctif ayant été approuvées à des fins de déploiement. Cette liste des mises à jour est envoyée à l'API Windows Update, qui détermine lesquelles des mises à jour sont applicables à l'instance et les installe au besoin. Si des mises à jour sont installées, l'instance est ensuite redémarrée autant de fois que nécessaire pour exécuter toutes les mises à jour correctives nécessaires.(Exception : si le paramètre `RebootOption` est défini sur `NoReboot` dans le document `AWS-RunPatchBaseline`, l'instance n'est pas redémarrée après l'exécution du Gestionnaire de correctifs. Pour de plus amples informations, veuillez consulter Nom du paramètre : `RebootOption` (p. 1087).) La synthèse de l'opération d'application de correctifs se situe dans la sortie de la demande Fonctionnalité Exécuter la commande. Des journaux supplémentaires sont disponibles sur l'instance dans le dossier `%PROGRAMDATA%\Amazon\PatchBaselineOperations\Logs`.

Du fait que l'API Windows Update est utilisée pour télécharger et installer des correctifs, tous les paramètres de Stratégie de groupe de Windows Update sont respectés. Aucun paramètre de stratégie de groupe n'est requis pour utiliser Gestionnaire de correctifs , mais tous les paramètres que vous avez définis seront appliqués, afin de diriger les instances vers un serveur WSUS (Windows Server Update Services).

> **Note**
>
> Par défaut, Windows télécharge tous les correctifs à partir du site Windows Update de Microsoft, car Gestionnaire de correctifs utilise l'API Windows Update pour guider le téléchargement et l'installation des correctifs. Par conséquent, l'instance doit être en mesure de joindre le site Windows Update de Microsoft, sinon l'application des correctifs échouera. Vous pouvez également configurer un serveur WSUS pour qu'il serve de référentiel de correctifs et configurer vos instances pour cibler ce serveur WSUS plutôt que d'utiliser les Stratégies de groupe.

Fonctionnement des règles de référence de correctif sur les systèmes basés sur Linux

Les règles d'une référence de correctif pour les distributions Linux fonctionnent différemment en fonction du type de distribution. Contrairement aux mises à jour de correctifs sur Windows Server les instances, les règles sont évaluées sur chaque instance pour prendre en compte les référentiels configurés sur l'instance. Gestionnaire de correctifs utilise le gestionnaire de package natif pour guider l'installation des correctifs approuvés par la référence de correctifs.

Rubriques

- Fonctionnement des règles de référence de correctifs sur Amazon Linux et Amazon Linux 2 (p. 1063)
- Fonctionnement des règles de référence de correctif sur CentOS (p. 1065)
- Fonctionnement des règles de référence de correctif sur Debian Server (p. 1067)
- Fonctionnement des règles de référence de correctif sur Oracle Linux (p. 1069)
- Fonctionnement des règles de référence de correctif sur RHEL (p. 1071)
- Fonctionnement des règles de référence de correctif sur SUSE Linux Enterprise Server (p. 1073)
- Fonctionnement des règles de référence de correctif sur Ubuntu Server (p. 1074)

Fonctionnement des règles de référence de correctifs sur Amazon Linux et Amazon Linux 2

Sur Amazon Linux et Amazon Linux 2, le processus de sélection de correctifs est le suivant :

1. Sur l'instance, la bibliothèque YUM accède au fichier `updateinfo.xml` pour chaque référentiel configuré.

   Note

   Si aucun fichier `updateinfo.xml` n'est trouvé, le correctif sera appliqué.

2. Chaque notice de mise à jour dans le fichier `updateinfo.xml` inclut plusieurs attributs indiquant les propriétés des packages de la notice, comme décrit dans le tableau suivant.

   Mettre à jour les attributs de la notice

Attribut	Description
type	Correspond à la valeur de l'attribut de la clé de classification du type de données PatchFilter de la référence de correctif. Indique le type de package inclus dans la notice de mise à jour. Vous pouvez afficher la liste des valeurs prises en charge à l'aide de la commande AWS CLI describe-patch-properties ou de l'action d'API DescribePatchProperties. Vous pouvez également afficher la liste dans la zone Règles d'approbation de la page Créer une référence de correctif ou Modifier la référence de correctif dans la console Systems Manager.
severity	Correspond à la valeur de l'attribut de la clé de sécurité du type de données PatchFilter de la référence de correctif. Indique l'importance des packages inclus dans la notice de mise à jour. En

Attribut	Description
	général, applicable uniquement aux notices de mise à jour de sécurité. Vous pouvez afficher la liste des valeurs prises en charge à l'aide de la commande AWS CLI describe-patch-properties ou de l'action d'API DescribePatchProperties. Vous pouvez également afficher la liste dans la zone Règles d'approbation de la page Créer une référence de correctif ou Modifier la référence de correctif dans la console Systems Manager.
update_id	Indique l'ID Advisory, tels que ALAS-2017-867. L'ID Advisory peut être utilisé dans l'attribut Correctifs approuvés ou RejectedPatches, dans la référence de correctifs.
references	Contient des informations supplémentaires sur la notice de mise à jour, notamment l'ID CVE (format : CVE-2017-1234567). L'ID CVE peut être utilisé dans l'attribut Correctifs approuvés ou RejectedPatches, dans la référence de correctifs.
updated	Correspond à ApproveAfterDays dans la référence de correctif. Indique la date de publication (date de mise à jour) des packages inclus dans la notice de mise à jour. Une comparaison entre l'horodatage actuel et la valeur de cet attribut plus le ApproveAfterDays est utilisée pour déterminer si le correctif est approuvé pour le déploiement.

Note

Pour plus d'informations sur les formats acceptés pour les listes de correctifs approuvés et de correctifs rejetés, consultez À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés (p. 1103).

3. Le produit de l'instance est déterminé par Agent SSM. Cet attribut correspond à la valeur de l'attribut de la clé de produit du type de données PatchFilter de la référence de correctif.
4. Les packages sont sélectionnés pour la mise à jour en suivant les consignes suivantes :

Option de sécurité	Sélection de correctifs
Les références de correctifs par défaut prédéfinies fournies par AWS et les références de correctifs personnalisés pour lesquelles l'option Approved patches include non-security updates (Les correctifs approuvés incluent des mises à jour non liées à la sécurité) n'est pas sélectionnée	Pour chaque notice de mise à jour dans `updateinfo.xml`, la référence de correctif est utilisée en guise de filtre, ce qui permet que seuls les packages qualifiés soient inclus dans la mise à jour. Si plusieurs packages sont applicables après l'application de la définition de référence de correctif, la version la plus récente est utilisée. La commande yum équivalente pour ce flux de travail est :

Option de sécurité	Sélection de correctifs
	`sudo yum update-minimal --sec-severity=critical,important --bugfix -y`
Références de correctifs personnalisées pour lesquelles l'option Approved patches include non-security updates (Les correctifs approuvés incluent des mises à jour non liées à la sécurité) est sélectionnée	En plus d'appliquer les mises à jour de sécurité qui ont été sélectionnées à partir de `updateinfo.xml` , applique Gestionnaire de correctifs les mises à jour non liées à la sécurité qui respectent par ailleurs les règles de filtrage des correctifs. La commande yum équivalente pour ce flux de travail est : `sudo yum update --security --bugfix -y`

Pour de plus amples informations sur les valeurs du statut de conformité des correctifs, veuillez consulter Présentation des valeurs d'état de conformité des correctifs (p. 1113).

Fonctionnement des règles de référence de correctif sur CentOS

Sur les CentOS, le processus de sélection des correctifs est le suivant :

1. Sur l'instance, la bibliothèque YUM (sur les versions CentOS 6.x et 7.x) ou la bibliothèque DNF (sur CentOS 8.x) accède au `updateinfo.xml` fichier pour chaque référentiel configuré.

   Note

   Si aucun fichier `updateinfo.xml` n'est trouvé, aucun correctif ne sera appliqué.

2. Chaque notice de mise à jour dans le fichier `updateinfo.xml` inclut plusieurs attributs indiquant les propriétés des packages de la notice, comme décrit dans le tableau suivant.

   Mettre à jour les attributs de la notice

Attribut	Description
type	Correspond à la valeur de l'attribut de la clé de classification du type de données PatchFilter de la référence de correctif. Indique le type de package inclus dans la notice de mise à jour. Vous pouvez afficher la liste des valeurs prises en charge à l'aide de la commande AWS CLI describe-patch-properties ou de l'action d'API DescribePatchProperties. Vous pouvez également afficher la liste dans la zone Règles d'approbation de la page Créer une référence de correctif ou Modifier la référence de correctif dans la console Systems Manager.
severity	Correspond à la valeur de l'attribut de la clé de gravité du type de données PatchFilter de la référence de correctif. Indique l'importance des packages inclus dans la notice de mise à jour. En

Attribut	Description
	général, applicable uniquement aux notices de mise à jour de sécurité. Vous pouvez afficher la liste des valeurs prises en charge à l'aide de la commande AWS CLI describe-patch-properties ou de l'action d'API DescribePatchProperties. Vous pouvez également afficher la liste dans la zone Règles d'approbation de la page Créer une référence de correctif ou Modifier la référence de correctif dans la console Systems Manager.
update_id	Indique l'ID Advisory, tels que CVE-2019-17055. L'ID Advisory peut être utilisé dans l'attribut Correctifs approuvés ou RejectedPatches, dans la référence de correctifs.
references	Contient des informations supplémentaires sur la notice de mise à jour, notamment l'ID CVE (format : CVE-2019-17055) ou un ID Bugzilla (format : 1463241). L'ID CVE et l'ID Bugzilla peuvent être utilisés dans l'attribut Correctifs approuvés ou RejectedPatches, dans la référence de correctifs.
updated	Correspond à ApproveAfterDays dans la référence de correctif. Indique la date de publication (date de mise à jour) des packages inclus dans la notice de mise à jour. Une comparaison entre l'horodatage actuel et la valeur de cet attribut plus le ApproveAfterDays est utilisée pour déterminer si le correctif est approuvé pour le déploiement.

Note

Pour plus d'informations sur les formats acceptés pour les listes de correctifs approuvés et de correctifs rejetés, consultez À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés (p. 1103).

3. Le produit de l'instance est déterminé par Agent SSM. Cet attribut correspond à la valeur de l'attribut de la clé de produit du type de données PatchFilter de la référence de correctif.
4. Les packages sont sélectionnés pour la mise à jour en suivant les consignes suivantes :

Option de sécurité	Sélection de correctifs
Les références de correctifs par défaut prédéfinies fournies par AWS et les références de correctifs personnalisés pour lesquelles l'option Approved patches include non-security updates (Les correctifs approuvés incluent des mises à jour non liées à la sécurité) n'est pas sélectionnée	Pour chaque notice de mise à jour dans `updateinfo.xml`, la référence de correctif est utilisée en guise de filtre, ce qui permet que seuls les packages qualifiés soient inclus dans la mise à jour. Si plusieurs packages sont applicables après l'application de la définition de référence de correctif, la version la plus récente est utilisée.

Option de sécurité	Sélection de correctifs

	Pour CentOS les versions 6 et 7, la commande yum équivalente pour ce flux de travail est :
	`sudo yum update-minimal --sec-severity=critical,important --bugfix -y`
	Pour CentOS 8, les commandes dnf équivalentes pour ce flux de travail sont les suivantes :
	`sudo dnf update-minimal --sec-severity=Critical --bugfix -y \`
	`sudo dnf update-minimal --sec-severity=Important --bugfix -y`

Références de correctifs personnalisées pour lesquelles l'option Approved patches include non-security updates (Les correctifs approuvés incluent des mises à jour non liées à la sécurité) est sélectionnée	En plus d'appliquer les mises à jour de sécurité qui ont été sélectionnées à partir de `updateinfo.xml` , applique Gestionnaire de correctifs les mises à jour non liées à la sécurité qui respectent par ailleurs les règles de filtrage des correctifs.
	Pour CentOS les versions 6 et 7, la commande yum équivalente pour ce flux de travail est :
	`sudo yum update --security --bugfix -y`
	Pour CentOS 8, la commande dnf équivalente pour ce flux de travail est :
	`sudo dnf update --security --bugfix -y`

Pour de plus amples informations sur les valeurs du statut de conformité des correctifs, veuillez consulter Présentation des valeurs d'état de conformité des correctifs (p. 1113).

Fonctionnement des règles de référence de correctif sur Debian Server

Sur Debian Server, le service de référence de correctifs permet un filtrage sur les champs Priority (Priorité) et Section. Ces champs sont généralement présents pour tous les packages Debian Server. Pour déterminer si un correctif est sélectionné par la référence de correctifs, Gestionnaire de correctifs effectue les opérations suivantes :

1. Sous les systèmes Debian Server, l'équivalent de `sudo apt-get update` est exécuté afin d'actualiser la liste des packages disponibles. Les référentiels ne sont pas configurés et les données sont extraites des référentiels configurés dans une liste de `sources`.

   > **Important**
   >
   > Sur Debian Server 8 uniquement : Étant donné que les systèmes d'exploitation Debian Server 8.* font référence à un référentiel de packages obsolète (`jessie-backports`), Gestionnaire de correctifs effectue les étapes supplémentaires suivantes pour s'assurer que les opérations d'application de correctifs réussissent :

a. Sur votre instance, la référence au référentiel `jessie-backports` est commentée à partir de la liste des emplacements source (`/etc/apt/sources.list.d/jessie-backports`). Par conséquent, aucune tentative n'est effectuée pour télécharger des correctifs à partir de cet emplacement.

b. Une clé de signature de mise à jour de sécurité Stretch est importée. Cette clé fournit les autorisations nécessaires pour les opérations de mise à jour et d'installation sur les distributions Debian Server 8.*.

c. Une opération `apt-get` est exécutée pour s'assurer que la dernière version de `python3-apt` est installée avant le début du processus d'application des correctifs.

d. Une fois le processus d'installation terminé, la référence au référentiel `jessie-backports` est restaurée et la clé de signature est supprimée du porte-clés source apt. Ainsi, la configuration du système demeure telle qu'elle était avant l'opération d'application de correctifs.

2. Ensuite, les listes GlobalFilters, ApprovalRules, Correctifs approuvés et RejectedPatches sont appliquées.

   Note

   Comme il n'est pas possible de déterminer de manière fiable les dates de publication des packages de mise à jour pour Debian Server, les options d'approbation automatique ne sont pas prises en charge pour ce système d'exploitation.

   Les règles d'approbation, toutefois, sont également soumises à l'activation ou non de la case à cocher Inclure les mises à jour non liées à la sécurité lors de la création ou de la dernière mise à jour d'une référence de correctifs.

   Si les mises à jour non liées à la sécurité sont exclues, une règle implicite est appliquée afin de sélectionner uniquement les packages avec des mises à niveau dans les référentiels de sécurité. Pour chaque package, la version du package proposée (généralement la plus récente) doit se trouver dans un référentiel de sécurité. Dans ce cas, pour Debian Server , les versions des correctifs candidats sont limitées aux correctifs inclus dans les référentiels suivants :

   Ces référentiels sont nommés comme suit :

   - Debian Server8 :`debian-security jessie`
   - Debian Server9 :`debian-security stretch`

   Si des mises à jour non liées à la sécurité sont incluses, les correctifs d'autres référentiels sont également pris en compte.

   Note

   Pour plus d'informations sur les formats acceptés pour les listes de correctifs approuvés et de correctifs rejetés, consultez À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés (p. 1103).

Pour afficher le contenu des champs Priorité et Section, exécutez la commande `aptitude` suivante :

Note

Vous devrez peut-être d'abord installer Aptitude sur les Debian Server systèmes .

```
aptitude search -F '%p %P %s %t %V#' '~U'
```

Dans la réponse à cette commande, tous les packages pouvant être mis à niveau sont indiqués dans le format suivant :

```
name, priority, section, archive, candidate version
```

Pour de plus amples informations sur les valeurs du statut de conformité des correctifs, veuillez consulter Présentation des valeurs d'état de conformité des correctifs (p. 1113).

Fonctionnement des règles de référence de correctif sur Oracle Linux

Sur les Oracle Linux, le processus de sélection des correctifs est le suivant :

1. Sur l'instance, la bibliothèque YUM accède au fichier `updateinfo.xml` pour chaque référentiel configuré.

   Note

   Le `updateinfo.xml` fichier peut ne pas être disponible si le référentiel n'est pas géré par Oracle. Si aucun correctif n'est `updateinfo.xml` trouvé, aucun correctif ne sera appliqué.

2. Chaque notice de mise à jour dans le fichier `updateinfo.xml` inclut plusieurs attributs indiquant les propriétés des packages de la notice, comme décrit dans le tableau suivant.

   Mettre à jour les attributs de la notice

Attribut	Description
type	Correspond à la valeur de l'attribut de la clé de classification du type de données PatchFilter de la référence de correctif. Indique le type de package inclus dans la notice de mise à jour. Vous pouvez afficher la liste des valeurs prises en charge à l'aide de la commande AWS CLI describe-patch-properties ou de l'action d'API DescribePatchProperties. Vous pouvez également afficher la liste dans la zone Règles d'approbation de la page Créer une référence de correctif ou Modifier la référence de correctif dans la console Systems Manager.
severity	Correspond à la valeur de l'attribut de la clé de gravité du type de données PatchFilter de la référence de correctif. Indique l'importance des packages inclus dans la notice de mise à jour. En général, applicable uniquement aux notices de mise à jour de sécurité. Vous pouvez afficher la liste des valeurs prises en charge à l'aide de la commande AWS CLI describe-patch-properties ou de l'action d'API DescribePatchProperties. Vous pouvez également afficher la liste dans la zone Règles d'approbation de la page Créer une référence de correctif ou Modifier la référence de correctif dans la console Systems Manager.
update_id	Indique l'ID Advisory, tels que CVE-2019-17055. L'ID Advisory peut être utilisé dans l'attribut Correctifs approuvés ou RejectedPatches, dans la référence de correctifs.

Attribut	Description
references	Contient des informations supplémentaires sur la notice de mise à jour, notamment l'ID CVE (format : CVE-2019-17055) ou un ID Bugzilla (format : 1463241). L'ID CVE et l'ID Bugzilla peuvent être utilisés dans l'attribut Correctifs approuvés ou RejectedPatches, dans la référence de correctifs.
updated	Correspond à ApproveAfterDays dans la référence de correctif. Indique la date de publication (date de mise à jour) des packages inclus dans la notice de mise à jour. Une comparaison entre l'horodatage actuel et la valeur de cet attribut plus le ApproveAfterDays est utilisée pour déterminer si le correctif est approuvé pour le déploiement.

> **Note**
>
> Pour plus d'informations sur les formats acceptés pour les listes de correctifs approuvés et de correctifs rejetés, consultez À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés (p. 1103).

3. Le produit de l'instance est déterminé par Agent SSM. Cet attribut correspond à la valeur de l'attribut de la clé de produit du type de données PatchFilter de la référence de correctif.
4. Les packages sont sélectionnés pour la mise à jour en suivant les consignes suivantes :

Option de sécurité	Sélection de correctifs
Les références de correctifs par défaut prédéfinies fournies par AWS et les références de correctifs personnalisés pour lesquelles l'option Approved patches include non-security updates (Les correctifs approuvés incluent des mises à jour non liées à la sécurité) n'est pas sélectionnée	Pour chaque notice de mise à jour dans `updateinfo.xml`, la référence de correctif est utilisée en guise de filtre, ce qui permet que seuls les packages qualifiés soient inclus dans la mise à jour. Si plusieurs packages sont applicables après l'application de la définition de référence de correctif, la version la plus récente est utilisée. La commande yum équivalente pour ce flux de travail est : `sudo yum update-minimal --sec-severity=important,moderate --bugfix -y`
Références de correctifs personnalisées pour lesquelles l'option Approved patches include non-security updates (Les correctifs approuvés incluent des mises à jour non liées à la sécurité) est sélectionnée	En plus d'appliquer les mises à jour de sécurité qui ont été sélectionnées à partir de `updateinfo.xml` , applique Gestionnaire de correctifs les mises à jour non liées à la sécurité qui respectent par ailleurs les règles de filtrage des correctifs. La commande yum équivalente pour ce flux de travail est :

Option de sécurité	Sélection de correctifs
	`sudo yum update --security --bugfix`

Pour de plus amples informations sur les valeurs du statut de conformité des correctifs, veuillez consulter Présentation des valeurs d'état de conformité des correctifs (p. 1113).

Fonctionnement des règles de référence de correctif sur RHEL

Sur les Red Hat Enterprise Linux, le processus de sélection des correctifs est le suivant :

1. Sur l'instance, la bibliothèque YUM (RHEL 7) ou la bibliothèque DNF (RHEL 8) accède au fichier `updateinfo.xml` de chaque référentiel configuré.

   Note

   Le fichier `updateinfo.xml` peut ne pas être disponible si le référentiel n'est pas géré par Red Hat. Si aucun fichier `updateinfo.xml` n'est trouvé, aucun correctif ne sera appliqué.

2. Chaque notice de mise à jour dans le fichier `updateinfo.xml` inclut plusieurs attributs indiquant les propriétés des packages de la notice, comme décrit dans le tableau suivant.

   Mettre à jour les attributs de la notice

Attribut	Description
type	Correspond à la valeur de l'attribut de la clé de classification du type de données PatchFilter de la référence de correctif. Indique le type de package inclus dans la notice de mise à jour. Vous pouvez afficher la liste des valeurs prises en charge à l'aide de la commande AWS CLI describe-patch-properties ou de l'action d'API DescribePatchProperties. Vous pouvez également afficher la liste dans la zone Règles d'approbation de la page Créer une référence de correctif ou Modifier la référence de correctif dans la console Systems Manager.
severity	Correspond à la valeur de l'attribut de la clé de gravité du type de données PatchFilter de la référence de correctif. Indique l'importance des packages inclus dans la notice de mise à jour. En général, applicable uniquement aux notices de mise à jour de sécurité. Vous pouvez afficher la liste des valeurs prises en charge à l'aide de la commande AWS CLI describe-patch-properties ou de l'action d'API DescribePatchProperties. Vous pouvez également afficher la liste dans la zone Règles d'approbation de la page Créer une référence de correctif ou Modifier la référence de correctif dans la console Systems Manager.

Attribut	Description
update_id	Indique l'ID Advisory, tels que RHSA-2017:0864. L'ID Advisory peut être utilisé dans l'attribut Correctifs approuvés ou RejectedPatches, dans la référence de correctifs.
references	Contient des informations supplémentaires sur la notice de mise à jour, notamment l'ID CVE (format : CVE-2017-1000371) ou un ID Bugzilla (format : 1463241). L'ID CVE et l'ID Bugzilla peuvent être utilisés dans l'attribut Correctifs approuvés ou RejectedPatches, dans la référence de correctifs.
updated	Correspond à ApproveAfterDays dans la référence de correctif. Indique la date de publication (date de mise à jour) des packages inclus dans la notice de mise à jour. Une comparaison entre l'horodatage actuel et la valeur de cet attribut plus le ApproveAfterDays est utilisée pour déterminer si le correctif est approuvé pour le déploiement.

Note

Pour plus d'informations sur les formats acceptés pour les listes de correctifs approuvés et de correctifs rejetés, consultez À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés (p. 1103).

3. Le produit de l'instance est déterminé par Agent SSM. Cet attribut correspond à la valeur de l'attribut de la clé de produit du type de données PatchFilter de la référence de correctif.
4. Les packages sont sélectionnés pour la mise à jour en suivant les consignes suivantes :

Option de sécurité	Sélection de correctifs

Les références de correctifs par défaut prédéfinies fournies par AWS et les références de correctifs personnalisés pour lesquelles l'option Approved patches include non-security updates (Les correctifs approuvés incluent des mises à jour non liées à la sécurité) n'est pas sélectionnée	Pour chaque notice de mise à jour dans `updateinfo.xml`, la référence de correctif est utilisée en guise de filtre, ce qui permet que seuls les packages qualifiés soient inclus dans la mise à jour. Si plusieurs packages sont applicables après l'application de la définition de référence de correctif, la version la plus récente est utilisée.
	
	Pour RHEL 7, la commande yum équivalente pour ce flux de travail est :
	
	`sudo yum update-minimal --sec-severity=critical,important --bugfix -y`
	
	Pour RHEL 8, les commandes yum équivalentes pour ce flux de travail sont les suivantes :
	
	`sudo dnf update-minimal --sec-severity=Critical --bugfix -y \`

Option de sécurité	Sélection de correctifs
	`sudo dnf update-minimal --sec-severity=Important --bugfix -y`

Références de correctifs personnalisées pour lesquelles l'option Approved patches include non-security updates (Les correctifs approuvés incluent des mises à jour non liées à la sécurité) est sélectionnée	En plus d'appliquer les mises à jour de sécurité qui ont été sélectionnées à partir de `updateinfo.xml` , applique Gestionnaire de correctifs les mises à jour non liées à la sécurité qui respectent par ailleurs les règles de filtrage des correctifs.
	
	Pour RHEL 7, la commande yum équivalente pour ce flux de travail est :
	
	`sudo yum update --security --bugfix`
	
	Pour RHEL 8, la commande dnf équivalente pour ce flux de travail est :
	
	`sudo dnf update --security --bugfix`

Pour de plus amples informations sur les valeurs du statut de conformité des correctifs, veuillez consulter Présentation des valeurs d'état de conformité des correctifs (p. 1113).

Fonctionnement des règles de référence de correctif sur SUSE Linux Enterprise Server

Sur SLES, chaque correctif inclut les attributs suivants, qui indiquent les propriétés des packages dans le correctif :

- Catégorie : Correspond à la valeur de l'attribut de la clé de classification du type de PatchFilter données de la référence de correctif. Indique le type de correctif inclus dans la notice de mise à jour.

  Vous pouvez afficher la liste des valeurs prises en charge à l'aide de la commande AWS CLI describe-patch-properties ou de l'action d'API DescribePatchProperties. Vous pouvez également afficher la liste dans la zone Règles d'approbation de la page Créer une référence de correctif ou Modifier la référence de correctif dans la console Systems Manager.
- Gravité : Correspond à la valeur de l'attribut de la clé de gravité du type de PatchFilter données de la référence de correctif. Désigne la gravité des correctifs.

  Vous pouvez afficher la liste des valeurs prises en charge à l'aide de la commande AWS CLI describe-patch-properties ou de l'action d'API DescribePatchProperties. Vous pouvez également afficher la liste dans la zone Règles d'approbation de la page Créer une référence de correctif ou Modifier la référence de correctif dans la console Systems Manager.

Le produit de l'instance est déterminé par Agent SSM. Cet attribut correspond à la valeur de l'attribut de la clé Product du type de données PatchFilter de la référence de correctifs.

Pour chaque correctif, la référence de correctif est utilisée en guise de filtre, ce qui permet que seuls les packages qualifiés soient inclus dans la mise à jour. Si plusieurs packages sont applicables après l'application de la définition de référence de correctif, la version la plus récente est utilisée.

> **Note**
>
> Pour plus d'informations sur les formats acceptés pour les listes de correctifs approuvés et de correctifs rejetés, consultez À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés (p. 1103).

Fonctionnement des règles de référence de correctif sur Ubuntu Server

Sur Ubuntu Server, le service de référence de correctifs permet un filtrage sur les champs Priority (Priorité) et Section. Ces champs sont généralement présents pour tous les packages Ubuntu Server. Pour déterminer si un correctif est sélectionné par la référence de correctifs, Gestionnaire de correctifs effectue les opérations suivantes :

1. Sous les systèmes Ubuntu Server, l'équivalent de `sudo apt-get update` est exécuté afin d'actualiser la liste des packages disponibles. Les référentiels ne sont pas configurés et les données sont extraites des référentiels configurés dans une liste de `sources`.
2. Ensuite, les listes GlobalFilters, ApprovalRules, Correctifs approuvés et RejectedPatches sont appliquées.

   > **Note**
   >
   > Comme il n'est pas possible de déterminer de manière fiable les dates de publication des packages de mise à jour pour Ubuntu Server, les options d'approbation automatique ne sont pas prises en charge pour ce système d'exploitation.

   Toutefois, les règles d'approbation sont également soumises à l'activation ou à la dernière case à cocher d'inclusion des mises à jour non liées à la sécurité lors de la création ou de la dernière mise à jour d'une référence de correctifs.

   Si les mises à jour non liées à la sécurité sont exclues, une règle implicite est appliquée afin de sélectionner uniquement les packages avec des mises à niveau dans les référentiels de sécurité. Pour chaque package, la version du package proposée (généralement la plus récente) doit se trouver dans un référentiel de sécurité. Dans ce cas, pour Ubuntu Server , les versions des correctifs candidats sont limitées aux correctifs inclus dans les référentiels suivants :

   - Ubuntu Server14 :`trusty-security`
   - Ubuntu Server16 :`xenial-security`
   - Ubuntu Server18 :`bionic-security`
   - Ubuntu Server20 :`focal-security`

   Si des mises à jour non liées à la sécurité sont incluses, les correctifs d'autres référentiels sont également pris en compte.

   > **Note**
   >
   > Pour plus d'informations sur les formats acceptés pour les listes de correctifs approuvés et de correctifs rejetés, consultez À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés (p. 1103).

Pour afficher le contenu des champs Priorité et Section, exécutez la commande `aptitude` suivante :

> **Note**
>
> Vous pouvez devoir installer Aptitude sur les systèmes Ubuntu Server 16.

```
aptitude search -F '%p %P %s %t %V#' '~U'
```

Dans la réponse à cette commande, tous les packages pouvant être mis à niveau sont indiqués dans le format suivant :

```
name, priority, section, archive, candidate version
```

Pour de plus amples informations sur les valeurs du statut de conformité des correctifs, veuillez consulter Présentation des valeurs d'état de conformité des correctifs (p. 1113).

Différences clés entre l'application de correctifs dans Windows et Linux

Cette rubrique décrit les différences importantes entre l'application de correctifs dans Linux et Windows.

> **Note**
>
> Pour corriger les instances Linux, vos instances doivent exécuter l'Agent SSM version 2.0.834.0 ou ultérieure.
> Une version mise à jour de l'Agent SSM est publiée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Si une version antérieure de l'agent est en cours d'exécution sur une instance, certains processus de l'Agent SSM peuvent échouer. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir l'Agent SSM à jour sur vos instances. Pour de plus amples informations, veuillez consulter Automatisation des mises à jour versAgent SSM (p. 109). Pour être alerté des mises à jour de Agent SSM, abonnez-vous à la page Agent SSM Release Notes sur GitHub.

Différence 1 : Évaluation des correctifs

Linux

Pour l'application de correctifs Linux, Systems Manager évalue les règles de référence de correctifs et la liste des correctifs approuvés et rejetés sur chaque instance gérée. Systems Manager doit évaluer l'application des correctifs sur chaque instance. En effet, le service récupère la liste des correctifs et mises à jour connus depuis les référentiels configurés sur l'instance.

Windows

Gestionnaire de correctifsutilise différents processus sur les instances gérées Windows et les instances gérées Linux afin d'évaluer quels correctifs doivent être présents. Pour l'application de correctifs Windows, Systems Manager évalue les règles de référence de correctifs et la liste des correctifs approuvés et rejetés directement dans le service. Cette action est possible en raison du fait que les correctifs Windows sont tirés d'un même référentiel (Windows Update).

Différence 2 : **`Not Applicable`**Correctifs

En raison du grand nombre de packages disponibles pour les systèmes d'exploitation Linux, Systems Manager ne signale aucun détail concernant les correctifs à l'état Ne s'applique pas. Un `Not Applicable` correctif est, par exemple, un correctif pour le logiciel Apache lorsque l'instance n'a pas Apache installé. signalSystems Managere le nombre de `Not Applicable` correctifs dans le récapitulatif, mais si vous appelez lDescribeInstancePatches'API pour une instance, les données renvoyées n'incluent pas les correctifs avec un état `Not Applicable` . Ce comportement est différent de Windows.

Différence 3 : Prise en charge du document SSM

Le document SSM `AWS-ApplyPatchBaseline` ne prend pas en charge les instances Linux. Pour appliquer les références de correctifs aux instances Windows Server et Linux, le document SSM recommandé est `AWS-RunPatchBaseline` . Pour plus d'informations, consultez À propos des documents SSM pour les instances d'application de correctifs (p. 1076) et À propos du Document SSM AWS-RunPatchBaseline (p. 1080) .

Différence 4 : Correctifs d'application

est principalement axé sur l'application de correctifs aux systèmes d'exploitation.Gestionnaire de correctifs Cependant, vous pouvez également utiliser Gestionnaire de correctifs pour appliquer des correctifs à certaines applications sur vos instances.

Linux

Sur les systèmes d'exploitation Linux, Gestionnaire de correctifs utilise les référentiels configurés pour les mises à jour et ne fait pas la différence entre les systèmes d'exploitation et les correctifs d'application. Vous pouvez utiliser Gestionnaire de correctifs pour définir les référentiels à partir desquels extraire les mises à jour. Pour plus d'informations, consultez la section Spécification d'un autre référentiel source de correctifs (Linux) (p. 1054).

Windows

Sur les instances Windows Server, vous pouvez appliquer les règles d'approbation, ainsi que les exceptions de correctifs Approuvé et Rejeté pour les applications publiées par Microsoft, telles que Microsoft Word 2011 et Microsoft Exchange Server 2016. Pour plus d'informations, consultez la section Utilisation de références de correctifs personnalisées (p. 1127).

À propos des opérations d'application de correctifs

Les rubriques de cette section fournissent des informations pour vous aider à comprendre comment le Gestionnaire de correctifs service fonctionne.

Les informations contenues dans les rubriques suivantes s'appliquent lorsque vous appliquez des correctifs à des instances selon une planification et à la demande.

Rubriques

- À propos des configurations d'application de correctifs (p. 1076)
- À propos des documents SSM pour les instances d'application de correctifs (p. 1076)

À propos des configurations d'application de correctifs

Une configuration d'application de correctifs définit une opération unique d'application de correctifs. La configuration spécifie les instances d'application de correctifs sur lesquelles la référence de correctifs doit être appliquée, le calendrier d'application des correctifs et, généralement, la fenêtre de maintenance à laquelle la configuration doit être associée.

Pour créer une configuration d'application de correctifs, utilisez la page Configure patching (Configurer l'application des correctifs). Cette page vous permet d'associer une configuration d'application des correctifs à une fenêtre de maintenance existante, de créer une nouvelle fenêtre de maintenance pour la configuration ou d'exécuter une opération d'application des correctifs manuelle unique sur un ensemble d'instances. Pour plus d'informations, consultez la section Création d'une configuration d'application de correctifs (console) (p. 1135).

À propos des documents SSM pour les instances d'application de correctifs

Cette rubrique décrit les huit documents SSM actuellement disponibles pour vous aider à appliquer les dernières mises à jour de sécurité à vos instances gérées.

Nous vous recommandons actuellement d'utiliser seulement quatre de ces documents dans vos opérations d'application de correctifs. Ensemble, ces quatre documents SSM vous fournissent une gamme complète

d'options d'application de correctifs à l'aide d' AWS Systems Manager . Trois de ces documents ont été publiés plus tard que les quatre documents SSM existants qu'ils remplacent et représentent des extensions ou des consolidations de fonctionnalités.

Les quatre documents SSM recommandés sont les suivants :

- AWS-ConfigureWindowsUpdate
- AWS-InstallWindowsUpdates
- AWS-RunPatchBaseline
- Propriété AWS-RunPatchBaselineAssociation pour

Les quatre documents SSM existants qui sont toujours disponibles pour une utilisation dans certaines régions AWS, mais risquent d'être obsolètes à l'avenir, comprennent :

- AWS-ApplyPatchBaseline
- AWS-FindWindowsUpdates
- AWS-InstallMissingWindowsUpdates
- AWS-InstallSpecificWindowsUpdates

Consultez les sections suivantes pour plus d'informations sur l'utilisation de ces documents SSM lors de vos opérations d'application de correctifs.

Rubriques

- Documents SSM recommandés pour les instances d'application de correctifs (p. 1077)
- Documents SSM existants pour les instances d'application de correctifs (p. 1079)
- À propos du Document SSM AWS-RunPatchBaseline (p. 1080)
- À propos du document SSM AWS-RunPatchBaselineAssociation (p. 1088)
- Exemple de scénario pour l'utilisation du InstallOverrideList paramètre dans AWS-RunPatchBaseline (p. 1097)

Documents SSM recommandés pour les instances d'application de correctifs

L'utilisation des quatre documents SSM suivants est recommandée dans vos opérations d'application de correctifs aux instances gérées.

Documents SSM recommandés

- AWS-ConfigureWindowsUpdate (p. 1077)
- AWS-InstallWindowsUpdates (p. 1078)
- AWS-RunPatchBaseline (p. 1078)
- Propriété AWS-RunPatchBaselineAssociation pour (p. 1079)

AWS-ConfigureWindowsUpdate

Prend en charge la configuration des fonctions de base de Windows Update et leur utilisation pour installer les mises à jour automatiquement (ou pour désactiver les mises à jour automatiques). Disponible dans toutes les régions AWS.

Ce document SSM invite Windows Update à télécharger et installer les mises à jour spécifiées et à redémarrer des instances en fonction de vos besoins. Utilisez ce document avec State Manager pour vous assurer que Windows Update conserve sa configuration. Vous pouvez également l'exécuter manuellement en utilisant Fonctionnalité Exécuter la commande afin de modifier la configuration Windows Update.

Les paramètres disponibles dans ce document prennent en charge la spécification d'une catégorie de mises à jour à installer (ou si les mises à jour automatiques doivent être désactivées), ainsi que la spécification du jour de la semaine et l'heure de la journée pour exécuter les opérations d'application des correctifs. Ce document SSM est particulièrement utile si vous n'avez pas besoin de contrôler strictement les mises à jour Windows et si vous n'avez pas besoin de collecter des informations de conformité.

Remplace les documents SSM existants :

- None (Aucun)

AWS-InstallWindowsUpdates

Installe les mises à jour sur une instance Windows Server. Disponible dans toutes les régions AWS.

Ce document SSM fournit des fonctionnalités élémentaires d'application de correctifs dans les cas où vous souhaitez installer une mise à jour spécifique (à l'aide de l'outil `Include Kbs`) ou souhaitez installer des correctifs avec des classifications ou des catégories spécifiques, mais n'avez pas besoin des informations de conformité des correctifs.

Remplace les documents SSM existants :

- AWS-FindWindowsUpdates
- AWS-InstallMissingWindowsUpdates
- AWS-InstallSpecificWindowsUpdates

Les trois documents existants réalisent différentes fonctions, mais vous pouvez obtenir les mêmes résultats en utilisant différents paramètres avec le nouveau document SSM AWS-InstallWindowsUpdates. La configuration des paramètres est décrite dans Documents SSM existants pour les instances d'application de correctifs (p. 1079).

AWS-RunPatchBaseline

Installe les correctifs sur les instances ou analyse les instances afin de déterminer s'il manque des correctifs qualifiés. Disponible dans toutes les régions AWS.

AWS-RunPatchBaseline vous permet de contrôler les approbations de correctifs à l'aide de la référence de correctifs actuellement spécifiée comme « par défaut » pour un type de système d'exploitation. Informations sur la conformité des correctifs que vous pouvez consulter à l'aide des outils de conformité Systems Manager. Ces outils vous fournissent des informations sur l'état de conformité des correctifs de vos instances, par exemple pour quelles instances il manque des correctifs et l'objectif de ces derniers. Lorsque vous utilisez `AWS-RunPatchBaseline` , les informations de conformité des correctifs sont enregistrées à l'aide de la commande `PutInventory` d'API. Pour les systèmes d'exploitation Linux, des informations de conformité sont fournies pour les correctifs à la fois par le référentiel source par défaut configuré sur une instance et par les autres référentiels source que vous spécifiez dans une référence de correctifs personnalisée. Pour en savoir plus sur les autres référentiels source, consultez Spécification d'un autre référentiel source de correctifs (Linux) (p. 1054). Pour plus d'informations sur les outils de conformité Systems Manager, consultez Conformité de la configuration AWS Systems Manager (p. 778).

Remplace les documents existants :

- AWS-ApplyPatchBaseline

Le document existant AWS-ApplyPatchBaseline s'applique uniquement aux instances Windows Server et ne fournit pas de prise en charge de l'application des correctifs. Le nouveau document AWS-RunPatchBaseline fournit le même support pour les systèmes Windows et Linux. La version 2.0.834.0 ou une version ultérieure de l'Agent SSM est requise pour pouvoir utiliser le document AWS-RunPatchBaseline.

Pour plus d'informations sur le document SSM AWS-RunPatchBaseline, consultez À propos du Document SSM AWS-RunPatchBaseline (p. 1080).

Propriété AWS-RunPatchBaselineAssociation pour

Installe les correctifs sur les instances ou analyse les instances afin de déterminer s'il manque des correctifs qualifiés. Disponible dans toutes les régions AWS commerciales.

AWS-RunPatchBaselineAssociation diffère d'AWS-RunPatchBaseline en ce sens qu'il prend en charge l'utilisation de balises pour identifier la référence de correctifs à utiliser avec un ensemble de cibles lorsqu'il s'exécute. En outre, les données de conformité des correctifs sont compilées en termes d'State Managerassociation spécifique. Les données de conformité des correctifs collectées lors de l'exécution d'AWS-RunPatchBaselineAssociation sont enregistrées à l'aide de la commande `PutComplianceItems` d'API au lieu de la `PutInventory` commande . Cela empêche les données de conformité qui ne sont pas associées à cette association particulière d'être remplacées.

Pour les systèmes d'exploitation Linux, des informations de conformité sont fournies pour les correctifs à la fois par le référentiel source par défaut configuré sur une instance et par les autres référentiels source que vous spécifiez dans une référence de correctifs personnalisée. Pour en savoir plus sur les autres référentiels source, consultez Spécification d'un autre référentiel source de correctifs (Linux) (p. 1054). Pour plus d'informations sur les outils de conformité Systems Manager, consultez Conformité de la configuration AWS Systems Manager (p. 778).

Remplace les documents existants :

- None (Aucun)

Pour plus d'informations sur le document SSM AWS-RunPatchBaselineAssociation, consultez À propos du document SSM AWS-RunPatchBaselineAssociation (p. 1088) .

Documents SSM existants pour les instances d'application de correctifs

Les quatre documents SSM suivants sont toujours disponibles pour les opérations d'application de correctifs dans certaines régions AWS. Toutefois, ils sont susceptibles de devenir obsolètes, c'est pourquoi nous vous déconseillons de les utiliser. Utilisez plutôt les documents décrits dans Documents SSM recommandés pour les instances d'application de correctifs (p. 1077).

Documents SSM existants

- AWS-ApplyPatchBaseline (p. 1079)
- AWS-FindWindowsUpdates (p. 1079)
- AWS-InstallMissingWindowsUpdates (p. 1080)
- AWS-InstallSpecificWindowsUpdates (p. 1080)

AWS-ApplyPatchBaseline

Prend en charge uniquement les instances Windows Server, mais n'inclut pas de prise en charge des correctifs d'applications qui se trouvent dans son document de remplacement, AWS-RunPatchBaseline. Non disponible dans les régions AWS lancées après août 2017.

> **Note**
>
> Le document de remplacement de ce document SSM, AWS-RunPatchBaseline, nécessite une version 2.0.834.0 ou ultérieure de l'Agent SSM. Vous pouvez utiliser le document AWS-UpdateSSMAgent pour mettre à jour les instances avec la dernière version de l'agent.

AWS-FindWindowsUpdates

Remplacé par AWS-InstallWindowsUpdates, qui peut effectuer toutes les mêmes actions. Non disponible dans les régions AWS lancées après avril 2017.

Pour obtenir le même résultat qu'avec le document SSM existant, utilisez la configuration de paramètres suivante avec le document de remplacement recommandé, AWS-InstallWindowsUpdates :

- `Action` = `Scan`
- `Allow Reboot` = `False`

AWS-InstallMissingWindowsUpdates

Remplacé par AWS-InstallWindowsUpdates, qui peut effectuer toutes les mêmes actions. Non disponible dans les régions AWS lancées après avril 2017.

Pour obtenir le même résultat qu'avec le document SSM existant, utilisez la configuration de paramètres suivante avec le document de remplacement recommandé, AWS-InstallWindowsUpdates :

- `Action` = `Install`
- `Allow Reboot` = `True`

AWS-InstallSpecificWindowsUpdates

Remplacé par AWS-InstallWindowsUpdates, qui peut effectuer toutes les mêmes actions. Non disponible dans les régions AWS lancées après avril 2017.

Pour obtenir le même résultat qu'avec le document SSM existant, utilisez la configuration de paramètres suivante avec le document de remplacement recommandé, AWS-InstallWindowsUpdates :

- `Action` = `Install`
- `Allow Reboot` = `True`
- `Include Kbs` = *`comma-separated list of KB articles`*

À propos du Document SSM AWS-RunPatchBaseline

AWS Systems Managerprend en charge un document SSM pour Gestionnaire de correctifs , `AWS-RunPatchBaseline` , qui effectue des opérations d'application de correctifs sur les instances pour les mises à jour liées à la sécurité et d'autres types de mises à jour. Lorsque le document est exécuté, il utilise la référence de correctifs actuellement spécifiée en tant que « valeur par défaut » pour un type de système d'exploitation. Vous pouvez utiliser ce document `AWS-RunPatchBaseline` pour appliquer des correctifs aux systèmes d'exploitation et aux applications. (Sous Windows, la prise en charge des applications est limitée aux mises à jour des applications Microsoft.)

Ce document prend en charge les instances Linux et Windows. Ce document effectue les actions correspondant à chaque plateforme.

> **Note**
>
> prend Gestionnaire de correctifs également en charge le SSM document hérité AWS-ApplyPatchBaseline. Toutefois, ce document prend en charge l'application de correctifs uniquement sur les instances Windows. Nous vous recommandons vivement d'utiliser plutôt AWS-RunPatchBaseline, car il prend en charge l'application de correctifs sur les instances Linux et Windows. La version 2.0.834.0 ou une version ultérieure de l'Agent SSM est requise pour pouvoir utiliser le document AWS-RunPatchBaseline.

Windows

Sur Windows Server les instances , le document AWS-RunPatchBaseline télécharge et appelle un PowerShell module, qui télécharge à son tour un instantané de la référence de correctif qui s'applique à l'instance. Cet instantané de la référence de correctif est transmis à l'API Windows Update qui contrôle le téléchargement et l'installation des correctifs approuvés selon les besoins.

Linux

Sur les instances Linux, le document AWS-RunPatchBaseline invoque un module Python qui, à son tour, télécharge un instantané de la référence de correctif qui s'applique à l'instance. Cet instantané de référence de correctif utilise les règles et listes de correctifs approuvés et bloqués définies pour diriger le gestionnaire de package correspondant pour chaque type d'instance :

- Amazon LinuxLes instances , Amazon Linux 2 , CentOS , et Oracle Linux 7 utilisent YUM.RHEL Pour les opérations YUM, Gestionnaire de correctifs nécessite Python 2.6 ou version ultérieure.
- Les instances RHEL 8 utilisent DNF. Pour les opérations DNF, Gestionnaire de correctifs nécessite Python 2 ou Python 3. (Aucune des deux versions n'est installée par défaut sur RHEL 8. Vous devez installer l'une ou l'autre manuellement.)
- Debian Serveret Ubuntu Server les instances utilisent APT. Pour les opérations APT, Gestionnaire de correctifs nécessite Python 3.
- Les instances SUSE Linux Enterprise Server utilisent Zypper. Pour les opérations Zypper, Gestionnaire de correctifs nécessite Python 2.6 ou version ultérieure.

Une fois que toutes les mises à jour approuvées et applicables ont été installées, avec les redémarrages effectués si nécessaire, les informations de conformité des correctifs sont générées sur une instance et rapportées à Gestionnaire de correctifs .

Note

Si le `RebootOption` paramètre est défini sur `NoReboot` dans le `AWS-RunPatchBaseline` document, l'instance n'est pas redémarrée après les Gestionnaire de correctifs exécutions. Pour plus d'informations, consultez la section Nom du paramètre : `RebootOption` (p. 1087).

Pour plus d'informations sur l'affichage des données de conformité des correctifs, consultez A propos de la conformité des correctifs (p. 783).

Paramètres AWS-RunPatchBaseline

AWS-RunPatchBaseline prend en charge quatre paramètres. L' `Operation` Le paramètre est obligatoire. L' `InstallOverrideList` et `RebootOption` Les paramètres sont facultatifs. `Snapshot-ID` est techniquement facultatif, mais nous vous recommandons de lui fournir une valeur personnalisée lorsque vous exécutez AWS-RunPatchBaseline en dehors d'une fenêtre de maintenance, et de laisser Gestionnaire de correctifs fournir la valeur automatiquement lorsque le document est exécuté dans le cadre d'une opération de fenêtre de maintenance.

Parameters

- Nom du paramètre : Operation (p. 1081)
- Nom du paramètre : Snapshot ID (p. 1082)
- Nom du paramètre : InstallOverrideList (p. 1083)
- Nom du paramètre : RebootOption (p. 1087)

Nom du paramètre : `Operation`

Utilisation : Obligatoire.

Options : `Scan|Install`

Analyser

Lorsque vous choisissez cette `Scan` option, AWS-RunPatchBaseline détermine l'état de conformité des correctifs de l'instance et rapporte ces informations à Gestionnaire de correctifs . `Scan`'invite

pas à installer les mises à jour ou à redémarrer les instances. Cet opération identifie plutôt à quel emplacement il manque des mises à jour approuvées et applicables à l'instance.

Install

Lorsque vous choisissez l'option `Install`, AWS-RunPatchBaseline tente d'installer les mises à jour approuvées et applicables qui sont manquantes dans l'instance. Les informations sur la conformité des correctifs générées dans le cadre d'une opération `Install` ne répertorient pas les mises à jour manquantes, mais peuvent signaler les mises à jour avec un état d'échec si l'installation de la mise à jour a échoué pour une raison ou pour une autre. À chaque installation d'une mise à jour sur une instance, cette dernière est redémarrée pour s'assurer que la mise à jour est non seulement installée, mais également active. (Exception : si le paramètre `RebootOption` est défini sur `NoReboot` dans le document `AWS-RunPatchBaseline`, l'instance n'est pas redémarrée après l'exécution du Gestionnaire de correctifs. Pour de plus amples informations, veuillez consulter Nom du paramètre : `RebootOption` (p. 1087).)

> **Note**
>
> Si un correctif spécifié par les règles de référence est installé avant la Gestionnaire de correctifs mise à jour de l'instance, le système peut ne pas redémarrer comme prévu. Cela peut se produire lorsqu'un correctif est installé manuellement par un utilisateur ou installé automatiquement par un autre programme, tel que le package `unattended-upgrades` sur Ubuntu Server.

Nom du paramètre : `Snapshot ID`

Utilisation : Facultatif.

`Snapshot ID` est un ID unique (GUID) utilisé par Gestionnaire de correctifs pour s'assurer qu'un ensemble d'instances corrigées en une seule opération possèdent exactement le même ensemble de correctifs approuvés. Bien que le paramètre soit défini comme facultatif, nous recommandons deux bonnes pratiques différentes : l'une si vous exécutez AWS-RunPatchBaseline dans une fenêtre de maintenance, l'autre si l'exécution a lieu hors d'une fenêtre de maintenance, comme décrit dans le tableau ci-dessous.

Bonnes pratiques AWS-RunPatchBaseline

Mode	Bonne pratique	Details (Détails)
Exécution de AWS-RunPatchBaseline dans une fenêtre de maintenance	Ne fournissez pas d'ID d'instantané. le fournira pour vous.Gestionnaire de correctifs	Si vous utilisez une fenêtre de maintenance pour exécuter AWS-RunPatchBaseline, vous ne devriez pas fournir votre propre ID d'instantané généré. Dans ce scénario, Systems Manager fournit une valeur de GUID en fonction de l'ID d'exécution de la fenêtre de maintenance. Cela permet de garantir que l'ID correct est utilisé pour tous les appels de AWS-RunPatchBaseline dans cette fenêtre de maintenance. Si vous spécifiez une valeur dans ce scénario, notez que l'instantané de la référence de correctif peut ne pas rester en place pendant plus de 24 heures. Par la suite, un nouvel instantané

Mode	Bonne pratique	Details (Détails)
		est généré même si vous spécifiez le même ID après l'expiration de l'instantané.
Exécution de AWS-RunPatchBaseline hors d'une fenêtre de maintenance	Générez et spécifiez une valeur de GUID personnalisée pour l'ID d'instantané.[1]	Si vous n'utilisez pas de fenêtre de maintenance pour exécuter AWS-RunPatchBaseline, nous vous recommandons de générer et de spécifier un ID d'instantané unique pour chaque référence de correctif, en particulier si vous exécutez le document AWS-RunPatchBaseline sur plusieurs instances de la même opération. Si vous ne spécifiez pas d'ID dans ce scénario, Systems Manager génère un ID d'instantané différent pour chaque instance à laquelle la commande est envoyée. Cela peut entraîner la spécification d'ensembles de correctifs différents entre les instances. Par exemple, si vous exécutez le document AWS-RunPatchBaseline directement via la fonctionnalité Fonctionnalité Exécuter la commande et que vous ciblez un groupe de 50 instances. La spécification d'un ID de référence personnalisé entraîne la génération d'un instantané de référence unique qui permet d'évaluer et de corriger toutes les instances, en garantissant qu'elles se retrouvent dans un état cohérent.
[1] Vous pouvez utiliser n'importe quel outil capable de générer un GUID afin de générer une valeur pour le paramètre d'ID d'instantané. Par exemple, dans PowerShell , vous pouvez utiliser l'`New-Guid`applet de commande pour générer un GUID au format de `12345699-9405-4f69-bc5e-9315aEXAMPLE` .		

Nom du paramètre : `InstallOverrideList`

Utilisation : Facultatif.

`InstallOverrideList` vous permet de spécifier une URL https ou une URL Amazon Simple Storage Service de type chemin (Amazon S3) vers une liste de correctifs à installer. Cette liste d'installation de correctifs que vous conservez au format YAML remplace les correctifs spécifiés par la référence de correctifs par défaut actuelle. Cela vous offre un contrôle plus précis sur les correctifs installés sur vos instances.

Sachez que les rapports de conformité reflètent les états des correctifs en fonction de ce qui est spécifié dans la référence de correctifs et non pas de ce que vous spécifiez dans un `InstallOverrideList`

Liste des correctifs. En d'autres termes, les opérations d'analyse ignorent le `InstallOverrideList` Paramètre . Cela permet de garantir que les rapports de conformité reflètent constamment les états de correctif en fonction de la stratégie plutôt que de ce qui a été approuvé pour une opération spécifique d'application de correctifs.

Pour obtenir une description de la façon dont vous pouvez utiliser le paramètre `InstallOverrideList` pour appliquer différents types de correctifs à un groupe cible, selon des calendriers de fenêtre de maintenance différents, tout en utilisant une ligne de base de correctifs unique, veuillez consulter Exemple de scénario pour l'utilisation du InstallOverrideList paramètre dans AWS-RunPatchBaseline (p. 1097).

Formats d'URL valides

- Format des URL https :

```
https://s3.amazonaws.com/DOC-EXAMPLE-BUCKET/my-windows-override-list.yaml
```

- URL de style chemin Amazon S3 :

```
s3://DOC-EXAMPLE-BUCKET/my-windows-override-list.yaml
```

Formats de contenu YAML valides

Les formats que vous utilisez pour spécifier des correctifs dans votre liste dépendent du système d'exploitation de votre instance. Le format général, toutefois, est le suivant :

```
patches:
    -
        id: '{patch-d}'
        title: '{patch-title}'
        {additional-fields}:{values}
```

Vous pouvez fournir des champs supplémentaires dans votre fichier YAML, mais ils sont ignorés pendant les opérations d'application de correctifs.

De plus, nous vous recommandons de vérifier que le format de votre fichier YAML est valide avant d'ajouter ou de mettre à jour la liste dans votre compartiment S3. Pour plus d'informations sur le format YAML, consultez yaml.org. Pour les options de l'outil de validation, recherchez « validateurs de format yaml » sur le web.

- Linux

  id

  Le champ id est obligatoire. Utilisez-le pour spécifier des correctifs à l'aide du nom du package et de l'architecture. Exemple : `'dhclient.x86_64'`. Vous pouvez utiliser des caractères génériques dans l'ID pour indiquer plusieurs packages. Exemple : `'dhcp*'`et `'dhcp*1.*'` .

  Title

  Le champ title (titre) est facultatif mais, sur les systèmes Linux, il fournit des fonctionnalités de filtrage supplémentaires. Si vous utilisez le champ title (titre), il doit contenir les informations de version de package dans l'un des formats suivants :

  YUM/SUSE Linux Enterprise Server (SLES) :

  ```
  {name}.{architecture}:{epoch}:{version}-{release}
  ```

  APT

```
{name}.{architecture}:{version}
```

Pour les titres de correctifs Linux, vous pouvez utiliser un ou plusieurs caractères génériques dans n'importe quelle position pour étendre le nombre de correspondances de package. Exemple : `'*32:9.8.2-0.*.rc1.57.amzn1'`.

Exemple :

- La version 1.2.25 du package apt est actuellement installée sur votre instance, mais la version 1.2.27 est désormais disponible.
- Vous ajoutez apt.amd64 version 1.2.27 à la liste des correctifs. Elle dépend de apt utils.amd64 version 1.2.27, mais apt-utils.amd64 version 1.2.25 est spécifié dans la liste.

Dans ce cas, apt version 1.2.27 sera bloqué pour l'installation et signalé comme « Failed-NonCompliant ».

- Microsoft Windows

  id

  Le champ id est obligatoire. Utilisez-le pour spécifier des correctifs à l'aide de la base de connaissances Microsoft IDs (par exemple, KB2736693) et du bulletin de sécurité Microsoft IDs (par exemple, MS17-023).

  Tous les autres champs que vous voulez fournir dans une liste de correctifs pour Windows sont facultatifs et fournis à titre d'information uniquement. Vous pouvez utiliser des champs supplémentaires, tels que title, classification, severity ou autre, pour fournir des informations plus détaillées sur les correctifs spécifiés.

Autres champs

Tous les autres champs que vous voulez fournir dans une liste de correctifs pour Linux sont facultatifs et fournis à titre d'information uniquement. Vous pouvez utiliser des champs supplémentaires, tels que classification, severity ou autre, pour fournir des informations plus détaillées sur les correctifs spécifiés.

Exemples de listes de correctifs

- Amazon Linux

```
patches:
    -
        id: 'kernel.x86_64'
    -
        id: 'bind*.x86_64'
        title: '32:9.8.2-0.62.rc1.57.amzn1'
    -
        id: 'glibc*'
    -
        id: 'dhclient*'
        title: '*12:4.1.1-53.P1.28.amzn1'
    -
        id: 'dhcp*'
        title: '*10:3.1.1-50.P1.26.amzn1'
```

- CentOS

```
patches:
    -
        id: 'kernel.x86_64'
```

```
    -
        id: 'bind*.x86_64'
        title: '32:9.8.2-0.62.rc1.57.amzn1'
    -
        id: 'glibc*'
    -
        id: 'dhclient*'
        title: '*12:4.1.1-53.P1.28.amzn1'
    -
        id: 'dhcp*'
        title: '*10:3.1.1-50.P1.26.amzn1'
```

- Debian Server

```
patches:
    -
        id: 'apparmor.amd64'
        title: '2.10.95-0ubuntu2.9'
    -
        id: 'cryptsetup.amd64'
        title: '*2:1.6.6-5ubuntu2.1'
    -
        id: 'cryptsetup-bin.*'
        title: '*2:1.6.6-5ubuntu2.1'
    -
        id: 'apt.amd64'
        title: '*1.2.27'
    -
        id: 'apt-utils.amd64'
        title: '*1.2.25'
```

- Oracle Linux

```
patches:
    -
        id: 'audit-libs.x86_64'
        title: '*2.8.5-4.el7'
    -
        id: 'curl.x86_64'
        title: '*.el7'
    -
        id: 'grub2.x86_64'
        title: 'grub2.x86_64:1:2.02-0.81.0.1.el7'
    -
        id: 'grub2.x86_64'
        title: 'grub2.x86_64:1:*-0.81.0.1.el7'
```

- Red Hat Enterprise Linux (RHEL)

```
patches:
    -
        id: 'NetworkManager.x86_64'
        title: '*1:1.10.2-14.el7_5'
    -
        id: 'NetworkManager-*.x86_64'
        title: '*1:1.10.2-14.el7_5'
    -
        id: 'audit.x86_64'
        title: '*0:2.8.1-3.el7'
    -
        id: 'dhclient.x86_64'
        title: '*.el7_5.1'
    -
        id: 'dhcp*.x86_64'
```

```
        title: '*12:5.2.5-68.el7'
```

- SUSE Linux Enterprise Server (SLES)

```
patches:
    -
        id: 'amazon-ssm-agent.x86_64'
    -
        id: 'binutils'
        title: '*0:2.26.1-9.12.1'
    -
        id: 'glibc*.x86_64'
        title: '*2.19*'
    -
        id: 'dhcp*'
        title: '0:4.3.3-9.1'
    -
        id: 'lib*'
```

- Ubuntu Server

```
patches:
    -
        id: 'apparmor.amd64'
        title: '2.10.95-0ubuntu2.9'
    -
        id: 'cryptsetup.amd64'
        title: '*2:1.6.6-5ubuntu2.1'
    -
        id: 'cryptsetup-bin.*'
        title: '*2:1.6.6-5ubuntu2.1'
    -
        id: 'apt.amd64'
        title: '*1.2.27'
    -
        id: 'apt-utils.amd64'
        title: '*1.2.25'
```

- Windows

```
patches:
    -
        id: 'KB4284819'
        title: '2018-06 Cumulative Update for Windows Server 2016 (1709) for x64-based
 Systems (KB4284819)'
    -
        id: 'KB4284833'
    -
        id: 'KB4284835'
        title: '2018-06 Cumulative Update for Windows Server 2016 (1803) for x64-based
 Systems (KB4284835)'
    -
        id: 'KB4284880'
    -
        id: 'KB4338814'
```

Nom du paramètre : `RebootOption`

Utilisation : Facultatif.

Options : `RebootIfNeeded|NoReboot`

RebootIfNeeded

Lorsque vous choisissez cette `RebootIfNeeded` option, l'instance est redémarrée si de nouveaux correctifs sont Gestionnaire de correctifs installés ou si elle a détecté des correctifs avec un statut au `INSTALLED_PENDING_REBOOT` cours de l'`Install`opération. L'état `INSTALLED_PENDING_REBOOT` peut signifier que l'option `NoReboot` a été sélectionnée lors de la dernière exécution de l'opération `Install`. (Les correctifs installés en dehors d' ne Gestionnaire de correctifs reçoivent jamais le statut `INSTALLED_PENDING_REBOOT` .)

> **Note**
>
> Lorsque vous choisissez l'`RebootIfNeeded`option, Gestionnaire de correctifs n'évalue pas si un redémarrage est requis par le correctif. Un redémarrage se produit chaque fois qu'il manque des paquets ou des paquets dont l'état est `INSTALLED_PENDING_REBOOT`.

NoReboot

Lorsque vous choisissez l'`NoReboot`option, Gestionnaire de correctifs ne redémarre pas une instance, même si des correctifs ont été installés au cours de l'`Install`opération. Cette option est utile si vous savez que vos instances ne nécessitent pas de redémarrage après l'application des correctifs, ou si vous avez des applications ou des processus en cours d'exécution sur une instance qui ne doivent pas être perturbés par un redémarrage suite à une opération d'application de correctifs. Elle est également utile lorsque vous voulez plus de contrôle sur le timing des redémarrages d'instance, par exemple en utilisant une fenêtre de maintenance.

> **Note**
>
> Si vous choisissez l'`NoReboot`option et qu'un correctif est installé, le correctif se voit attribuer l'état `InstalledPendingReboot` . L'instance elle-même, toutefois, est marquée comme `Non-Compliant` . Après un redémarrage et l'exécution d`Scan`'une opération, l'état de l'instance est mis à jour sur `Compliant` .

Fichier de suivi d'installation des correctifs : Pour suivre l'installation des correctifs, en particulier les correctifs qui ont été installés depuis le dernier redémarrage du système, Systems Manager gère un fichier sur l'instance gérée.

> **Important**
>
> Ne supprimez pas ou ne modifiez pas le fichier de suivi. Si ce fichier est supprimé ou endommagé, le rapport de conformité des correctifs pour l'instance est inexact. Si cela se produit, redémarrez l'instance et exécutez une opération d'analyse des correctifs pour restaurer le fichier.

Ce fichier de suivi est stocké dans les emplacements suivants sur vos instances gérées :

- Systèmes d'exploitation Linux : `/var/log/amazon/ssm/patch-configuration/patch-states-configuration.json`
- Système d'exploitation Windows Server : `C:\ProgramData\Amazon\PatchBaselineOperations\State\PatchStatesConfiguration.json`

À propos du document SSM AWS-RunPatchBaselineAssociation

A l'instar du `AWS-RunPatchBaseline` document, `AWS-RunPatchBaselineAssociation` effectue des opérations d'application de correctifs sur les instances pour les mises à jour liées à la sécurité et d'autres types de mises à jour. Vous pouvez également utiliser le document `AWS-RunPatchBaselineAssociation` pour appliquer des correctifs pour les systèmes d'exploitation et les

applications. (Sur Windows Server, la prise en charge des applications est limitée à des mises à jour pour les applications Microsoft.)

> **Note**
>
> `AWS-RunPatchBaselineAssociation`n'est actuellement pas pris en charge pour les serveurs et les machines virtuelles (VM) sur site dans un environnement hybride.

Ce document prend en charge les instances EC2 pour Linux et Windows. Le document effectue les actions appropriées pour chaque plateforme, en appelant un module Python sur les instances Linux et un PowerShell module sur les instances Windows.

`AWS-RunPatchBaselineAssociation`, toutefois, diffère de ce qui suit :`AWS-RunPatchBaseline`

- Lorsque vous utilisez le `AWS-RunPatchBaselineAssociation` document, vous pouvez spécifier une paire de clés de balise dans le champ de `BaselineTags` paramètre du document. Si une référence de correctifs personnalisée de votre compte partage ces balises, Gestionnaire de correctifs utilise la référence balisée lorsqu'elle s'exécute sur les instances cibles au lieu de la référence de correctifs « par défaut » actuellement spécifiée pour le type de système d'exploitation.

  Les deux formats suivants sont valides pour votre `BaselineTags` paramètre :

  `Key=`*`tag-key`*`,Values=`*`tag-value`*

  `Key=`*`tag-key`*`,Values=`*`tag-value1`*`,`*`tag-value2`*`,`*`tag-value3`*
- Lorsque `AWS-RunPatchBaselineAssociation` s'exécute, les données de conformité des correctifs qu'il collecte sont enregistrées à l'aide de la commande `PutComplianceItems` d'API au lieu de la `PutInventory` commande , qui est utilisée par `AWS-RunPatchBaseline` . Cette différence signifie que les informations de conformité des correctifs sont stockées et signalées par une association spécifique. Les données de conformité des correctifs générées en dehors de cette association ne sont pas remplacées.
- Les informations de conformité des correctifs rapportées après l'exécution `AWS-RunPatchBaselineAssociation` indiquent si une instance est conforme ou non. Il n'inclut pas les détails au niveau du correctif, comme illustré par la sortie de la AWS CLI commande suivante. Notez que la commande filtre sur `Assocation` comme type de conformité :

```
aws ssm list-compliance-items \
    --resource-ids "i-02573cafcfEXAMPLE" \
    --resource-types "ManagedInstance" \
    --filters "Key=ComplianceType,Values=Association,Type=EQUAL" \
    --region us-east-2
```

  Le système renvoie des informations similaires à celles qui suivent :

```
{
    "ComplianceItems": [
        {
            "Status": "NON_COMPLIANT",
            "Severity": "UNSPECIFIED",
            "Title": "MyPatchAssociation",
            "ResourceType": "ManagedInstance",
            "ResourceId": "i-02573cafcfEXAMPLE",
            "ComplianceType": "Association",
            "Details": {
                "DocumentName": "AWS-RunPatchBaselineAssociation",
                "PatchBaselineId": "pb-0c10e65780EXAMPLE",
                "DocumentVersion": "1"
            },
            "ExecutionSummary": {
```

```
                "ExecutionTime": 1590698771.0
            },
            "Id": "3e5d5694-cd07-40f0-bbea-040e6EXAMPLE"
        }
    ]
}
```

Si une valeur de paire de clés de balise a été spécifiée comme paramètre pour le `AWS-RunPatchBaselineAssociation` document, Gestionnaire de correctifs recherche une référence de correctifs personnalisée qui correspond au type du système d'exploitation et a été balisée avec cette même paire de clés de balise. Cette recherche n'est pas limitée à la référence de correctifs par défaut actuelle ou à la référence attribuée à un groupe de correctifs. Si aucune référence n'est trouvée avec les balises spécifiées, recherche Gestionnaire de correctifs ensuite un groupe de correctifs, s'il a été spécifié dans la commande qui exécute `AWS-RunPatchBaselineAssociation` . Si aucun groupe de correctifs n'est associé, Gestionnaire de correctifs revient à la référence de correctifs par défaut actuelle pour le compte du système d'exploitation.

Si plusieurs références de correctifs sont trouvées avec les balises spécifiées dans le `AWS-RunPatchBaselineAssociation` document, Gestionnaire de correctifs renvoie un message d'erreur indiquant qu'une seule référence de correctifs peut être balisée avec cette paire clé-valeur pour que l'opération se poursuive.

> **Note**
>
> Sur les instances Linux, le gestionnaire de package approprié pour chaque type d'instance est utilisé pour installer les packages :
>
> - Amazon LinuxLes Amazon Linux 2 instances , CentOS , Oracle Linux , et utilisent YUM.RHEL Pour les opérations YUM, Gestionnaire de correctifs nécessite Python 2.6 ou version ultérieure.
> - Debian Serveret Ubuntu Server les instances utilisent APT. Pour les opérations APT, Gestionnaire de correctifs nécessite Python 3.
> - Les instances SUSE Linux Enterprise Server utilisent Zypper. Pour les opérations Zypper, Gestionnaire de correctifs nécessite Python 2.6 ou version ultérieure.

Une fois qu'une analyse est terminée, ou une fois que toutes les mises à jour approuvées et applicables ont été installées, avec des redémarrages effectués si nécessaire, des informations de conformité des correctifs sont générées sur une instance et rapportées au service de conformité des correctifs.

> **Note**
>
> Si le `RebootOption` paramètre est défini sur `NoReboot` dans le `AWS-RunPatchBaselineAssociation` document, l'instance n'est pas redémarrée après les Gestionnaire de correctifs exécutions. Pour plus d'informations, consultez la section Nom du paramètre : `RebootOption` (p. 1096).

Pour plus d'informations sur l'affichage des données de conformité des correctifs, consultez A propos de la conformité des correctifs (p. 783).

Paramètres AWS-RunPatchBaselineAssociation

AWS-RunPatchBaselineAssociation prend en charge quatre paramètres. L' `Operation` et `AssociationId` Les paramètres sont obligatoires. L' `InstallOverrideList`, `RebootOption`, et `BaselineTags` Les paramètres sont facultatifs.

Parameters

- Nom du paramètre : Operation (p. 1091)

- Nom du paramètre : BaselineTags (p. 1091)
- Nom du paramètre : AssociationId (p. 1092)
- Nom du paramètre : InstallOverrideList (p. 1092)
- Nom du paramètre : RebootOption (p. 1096)

Nom du paramètre : `Operation`

Utilisation : Obligatoire.

Options : `Scan|Install`

Analyser

Lorsque vous choisissez cette `Scan` option, AWS-RunPatchBaselineAssociation détermine l'état de conformité des correctifs de l'instance et rapporte ces informations à Gestionnaire de correctifs . `Scan`n'invite pas à installer les mises à jour ou à redémarrer les instances. Cet opération identifie plutôt à quel emplacement il manque des mises à jour approuvées et applicables à l'instance.

Install

Lorsque vous choisissez cette `Install` option, AWS-RunPatchBaselineAssociation tente d'installer les mises à jour approuvées et applicables qui sont manquantes dans l'instance. Les informations sur la conformité des correctifs générées dans le cadre d'une opération `Install` ne répertorient pas les mises à jour manquantes, mais peuvent signaler les mises à jour avec un état d'échec si l'installation de la mise à jour a échoué pour une raison ou pour une autre. À chaque installation d'une mise à jour sur une instance, cette dernière est redémarrée pour s'assurer que la mise à jour est non seulement installée, mais également active. (Exception : si le `RebootOption` paramètre est défini sur `NoReboot` dans le `AWS-RunPatchBaselineAssociation` document, l'instance n'est pas redémarrée après les Gestionnaire de correctifs exécutions. Pour plus d'informations, consultez Nom du paramètre : `RebootOption` (p. 1096) .)

> **Note**
>
> Si un correctif spécifié par les règles de référence est installé avant la Gestionnaire de correctifs mise à jour de l'instance, le système peut ne pas redémarrer comme prévu. Cela peut se produire lorsqu'un correctif est installé manuellement par un utilisateur ou installé automatiquement par un autre programme, tel que le package `unattended-upgrades` sur Ubuntu Server.

Nom du paramètre : `BaselineTags`

Utilisation : Facultatif.

`BaselineTags` est une paire clé-valeur de balise unique que vous choisissez et affectez à une référence de correctifs personnalisée individuelle. Vous pouvez spécifier une ou plusieurs valeurs pour ce paramètre. Les deux formats suivants sont valides :

```
Key=tag-key,Values=tag-value
```

```
Key=tag-key,Values=tag-value1,tag-value2,tag-value3
```

L' `BaselineTags` La valeur est utilisée par Gestionnaire de correctifs pour s'assurer qu'un ensemble d'instances corrigées en une seule opération possède exactement le même ensemble de correctifs approuvés. Lorsque l'opération d'application de correctifs s'exécute, Gestionnaire de correctifs vérifie si une référence de correctifs pour le type de système d'exploitation est balisée avec la même paire clé-valeur que vous spécifiez pour `BaselineTags` . S'il existe une correspondance, cette référence de correctifs personnalisée est utilisée. S'il n'y a pas de correspondance, une référence de correctifs est identifiée en

fonction d'un groupe de correctifs spécifié pour l'opération d'application des correctifs. S'il n'y en a pas, la référence de correctifs prédéfinie AWS pour ce système d'exploitation est utilisée.

> **Note**
>
> Vous n'avez pas besoin de baliser vos instances avec cette paire clé-valeur.

Nom du paramètre : `AssociationId`

Utilisation : Obligatoire.

`AssociationId` est l'ID d'une State Manager association existante. Il est utilisé par Gestionnaire de correctifs pour ajouter des données de conformité à l'association spécifiée. En envoyant les résultats de l'application des correctifs en tant que données de conformité d'association au lieu des données de conformité d'inventaire, les informations de conformité d'inventaire existantes pour vos instances ne sont pas remplacées après une opération d'application des correctifs, ni pour toute autre associationIDs. Si vous n'avez pas encore d'association que vous souhaitez utiliser, vous pouvez en créer une en exécutant create-association la commande . Exemple :

Linux

```
aws ssm create-association \
    --name "AWS-RunPatchBaselineAssociation" \
    --association-name "MyPatchAssociation" \
    --targets
 "Key=instanceids,Values=[i-02573cafcfEXAMPLE,i-07782c72faEXAMPLE,i-07782c72faEXAMPLE]"
 \
    --parameters "Operation=Scan" \
    --schedule-expression "cron(0 */30 * * * ? *)" \
    --sync-compliance "MANUAL" \
    --region us-east-2
```

Windows

```
aws ssm create-association ^
    --name "AWS-RunPatchBaselineAssociation" ^
    --association-name "MyPatchAssociation" ^
    --targets
 "Key=instanceids,Values=[i-02573cafcfEXAMPLE,i-07782c72faEXAMPLE,i-07782c72faEXAMPLE]"
 ^
    --parameters "Operation=Scan" ^
    --schedule-expression "cron(0 */30 * * * ? *)" ^
    --sync-compliance "MANUAL" ^
    --region us-east-2
```

Nom du paramètre : `InstallOverrideList`

Utilisation : Facultatif.

`InstallOverrideList` vous permet de spécifier une URL https ou une URL Amazon Simple Storage Service de type chemin (Amazon S3) vers une liste de correctifs à installer. Cette liste d'installation de correctifs que vous conservez au format YAML remplace les correctifs spécifiés par la référence de correctifs par défaut actuelle. Cela vous offre un contrôle plus précis sur les correctifs installés sur vos instances.

Sachez que les rapports de conformité reflètent les états des correctifs en fonction de ce qui est spécifié dans la référence de correctifs et non pas de ce que vous spécifiez dans un `InstallOverrideList` Liste des correctifs. En d'autres termes, les opérations d'analyse ignorent le `InstallOverrideList` Paramètre . Cela permet de garantir que les rapports de conformité reflètent constamment les états de

correctif en fonction de la stratégie plutôt que de ce qui a été approuvé pour une opération spécifique d'application de correctifs.

Formats d'URL valides

- Format des URL https :

```
https://s3.amazonaws.com/DOC-EXAMPLE-BUCKET/my-windows-override-list.yaml
```

- URL de style chemin Amazon S3 :

```
s3://DOC-EXAMPLE-BUCKET/my-windows-override-list.yaml
```

Formats de contenu YAML valides

Les formats que vous utilisez pour spécifier des correctifs dans votre liste dépendent du système d'exploitation de votre instance. Le format général, toutefois, est le suivant :

```
patches:
    -
        id: '{patch-d}'
        title: '{patch-title}'
        {additional-fields}:{values}
```

Vous pouvez fournir des champs supplémentaires dans votre fichier YAML, mais ils sont ignorés pendant les opérations d'application de correctifs.

De plus, nous vous recommandons de vérifier que le format de votre fichier YAML est valide avant d'ajouter ou de mettre à jour la liste dans votre compartiment S3. Pour plus d'informations sur le format YAML, consultez yaml.org. Pour les options de l'outil de validation, recherchez « validateurs de format yaml » sur le web.

- Microsoft Windows

  id

  Le champ id est obligatoire. Utilisez-le pour spécifier des correctifs à l'aide de la base de connaissances Microsoft IDs (par exemple, KB2736693) et du bulletin de sécurité Microsoft IDs (par exemple, MS17-023).

  Tous les autres champs que vous voulez fournir dans une liste de correctifs pour Windows sont facultatifs et fournis à titre d'information uniquement. Vous pouvez utiliser des champs supplémentaires, tels que title, classification, severity ou autre, pour fournir des informations plus détaillées sur les correctifs spécifiés.
- Linux

  id

  Le champ id est obligatoire. Utilisez-le pour spécifier des correctifs à l'aide du nom du package et de l'architecture. Exemple : `'dhclient.x86_64'`. Vous pouvez utiliser des caractères génériques dans l'ID pour indiquer plusieurs packages. Exemple : `'dhcp*'`et `'dhcp*1.*'` .

  title

  Le champ title (titre) est facultatif mais, sur les systèmes Linux, il fournit des fonctionnalités de filtrage supplémentaires. Si vous utilisez le champ title (titre), il doit contenir les informations de version de package dans l'un des formats suivants :

YUM/SUSE Linux Enterprise Server (SLES) :

```
{name}.{architecture}:{epoch}:{version}-{release}
```

APT

```
{name}.{architecture}:{version}
```

Pour les titres de correctifs Linux, vous pouvez utiliser un ou plusieurs caractères génériques dans n'importe quelle position pour étendre le nombre de correspondances de package. Exemple : `'*32:9.8.2-0.*.rc1.57.amzn1'`.

Exemple :

- La version 1.2.25 du package apt est actuellement installée sur votre instance, mais la version 1.2.27 est désormais disponible.
- Vous ajoutez apt.amd64 version 1.2.27 à la liste des correctifs. Elle dépend de apt utils.amd64 version 1.2.27, mais apt-utils.amd64 version 1.2.25 est spécifié dans la liste.

Dans ce cas, apt version 1.2.27 sera bloqué pour l'installation et signalé comme « Failed-NonCompliant ».

Autres champs

Tous les autres champs que vous voulez fournir dans une liste de correctifs pour Linux sont facultatifs et fournis à titre d'information uniquement. Vous pouvez utiliser des champs supplémentaires, tels que classification, severity ou autre, pour fournir des informations plus détaillées sur les correctifs spécifiés.

Exemples de listes de correctifs

- Windows

```
patches:
    -
        id: 'KB4284819'
        title: '2018-06 Cumulative Update for Windows Server 2016 (1709) for x64-based
 Systems (KB4284819)'
    -
        id: 'KB4284833'
    -
        id: 'KB4284835'
        title: '2018-06 Cumulative Update for Windows Server 2016 (1803) for x64-based
 Systems (KB4284835)'
    -
        id: 'KB4284880'
    -
        id: 'KB4338814'
```

- APT

```
patches:
    -
        id: 'apparmor.amd64'
        title: '2.10.95-0ubuntu2.9'
    -
        id: 'cryptsetup.amd64'
        title: '*2:1.6.6-5ubuntu2.1'
    -
        id: 'cryptsetup-bin.*'
```

```
        title: '*2:1.6.6-5ubuntu2.1'
    -
        id: 'apt.amd64'
        title: '*1.2.27'
    -
        id: 'apt-utils.amd64'
        title: '*1.2.25'
```

- Amazon Linux

```
patches:
    -
        id: 'kernel.x86_64'
    -
        id: 'bind*.x86_64'
        title: '32:9.8.2-0.62.rc1.57.amzn1'
    -
        id: 'glibc*'
    -
        id: 'dhclient*'
        title: '*12:4.1.1-53.P1.28.amzn1'
    -
        id: 'dhcp*'
        title: '*10:3.1.1-50.P1.26.amzn1'
```

- Red Hat Enterprise Linux (RHEL)

```
patches:
    -
        id: 'NetworkManager.x86_64'
        title: '*1:1.10.2-14.el7_5'
    -
        id: 'NetworkManager-*.x86_64'
        title: '*1:1.10.2-14.el7_5'
    -
        id: 'audit.x86_64'
        title: '*0:2.8.1-3.el7'
    -
        id: 'dhclient.x86_64'
        title: '*.el7_5.1'
    -
        id: 'dhcp*.x86_64'
        title: '*12:5.2.5-68.el7'
```

- SUSE Linux Enterprise Server (SLES)

```
patches:
    -
        id: 'amazon-ssm-agent.x86_64'
    -
        id: 'binutils'
        title: '*0:2.26.1-9.12.1'
    -
        id: 'glibc*.x86_64'
        title: '*2.19*'
    -
        id: 'dhcp*'
        title: '0:4.3.3-9.1'
    -
        id: 'lib*'
```

- Ubuntu Server

```
patches:
    -
        id: 'apparmor.amd64'
        title: '2.10.95-0ubuntu2.9'
    -
        id: 'cryptsetup.amd64'
        title: '*2:1.6.6-5ubuntu2.1'
    -
        id: 'cryptsetup-bin.*'
        title: '*2:1.6.6-5ubuntu2.1'
    -
        id: 'apt.amd64'
        title: '*1.2.27'
    -
        id: 'apt-utils.amd64'
        title: '*1.2.25'
```

- Windows

```
patches:
    -
        id: 'KB4284819'
        title: '2018-06 Cumulative Update for Windows Server 2016 (1709) for x64-based
 Systems (KB4284819)'
    -
        id: 'KB4284833'
    -
        id: 'KB4284835'
        title: '2018-06 Cumulative Update for Windows Server 2016 (1803) for x64-based
 Systems (KB4284835)'
    -
        id: 'KB4284880'
    -
        id: 'KB4338814'
```

Nom du paramètre : `RebootOption`

Utilisation : Facultatif.

Options : `RebootIfNeeded`|`NoReboot`

RebootIfNeeded

Lorsque vous choisissez cette `RebootIfNeeded` option, l'instance est redémarrée si de nouveaux correctifs sont Gestionnaire de correctifs installés ou si elle a détecté des correctifs avec un statut au `INSTALLED_PENDING_REBOOT` cours de l'`Install`opération. L'état `INSTALLED_PENDING_REBOOT` peut signifier que l'option `NoReboot` a été sélectionnée lors de la dernière exécution de l'opération `Install`. (Les correctifs installés en dehors d' ne Gestionnaire de correctifs reçoivent jamais le statut `INSTALLED_PENDING_REBOOT` .)

> **Note**
>
> Lorsque vous choisissez l'`RebootIfNeeded`option, Gestionnaire de correctifs n'évalue pas si un redémarrage est requis par le correctif. Un redémarrage se produit chaque fois qu'il manque des paquets ou des paquets dont l'état est `INSTALLED_PENDING_REBOOT`.

NoReboot

Lorsque vous choisissez l'`NoReboot`option, Gestionnaire de correctifs ne redémarre pas une instance, même si elle a installé des correctifs au cours de l'`Install`opération. Cette option est utile si vous savez que vos instances ne nécessitent pas de redémarrage après l'application des correctifs, ou si

vous avez des applications ou des processus en cours d'exécution sur une instance qui ne doivent pas être perturbés par un redémarrage suite à une opération d'application de correctifs. Elle est également utile lorsque vous voulez plus de contrôle sur le timing des redémarrages d'instance, par exemple en utilisant une fenêtre de maintenance.

Fichier de suivi de l'installation des correctifs : Pour suivre l'installation des correctifs, en particulier les correctifs qui ont été installés depuis le dernier redémarrage du système, Systems Manager gère un fichier sur l'instance gérée.

> **Important**
>
> Ne supprimez pas ou ne modifiez pas le fichier de suivi. Si ce fichier est supprimé ou endommagé, le rapport de conformité des correctifs pour l'instance est inexact. Si cela se produit, redémarrez l'instance et exécutez une opération d'analyse des correctifs pour restaurer le fichier.

Ce fichier de suivi est stocké dans les emplacements suivants sur vos instances gérées :

- Systèmes d'exploitation Linux : `/var/log/amazon/ssm/patch-configuration/patch-states-configuration.json`
- Système d'exploitation Windows Server : `C:\ProgramData\Amazon\PatchBaselineOperations\State\PatchStatesConfiguration.json`

Exemple de scénario pour l'utilisation du InstallOverrideList paramètre dans AWS-RunPatchBaseline

Vous pouvez utiliser l' `InstallOverrideList` lorsque vous souhaitez remplacer les correctifs spécifiés par la référence de correctifs par défaut actuelle. L'exemple suivant montre comment utiliser ce paramètre pour atteindre les objectifs suivants :

- Appliquez différents ensembles de correctifs à un groupe cible d'instances.
- Appliquez ces ensembles de correctifs sur différentes fréquences.
- Utilisez la même ligne de base de correctifs pour les deux opérations.

Supposons que vous souhaitez installer deux catégories différentes de correctifs sur vos instances Amazon Linux 2. Vous souhaitez installer ces correctifs sur différents calendriers à l'aide de fenêtres de maintenance. Vous voulez qu'une fenêtre de maintenance s'exécute chaque semaine et installe tous les correctifs `Security`. Vous souhaitez qu'une autre fenêtre de maintenance s'exécute une fois par mois et installe tous les correctifs disponibles ou catégories de correctifs autres que `Security`.

Toutefois, une seule ligne de base de correctifs à la fois peut être définie comme valeur par défaut pour un système d'exploitation. Cette exigence permet d'éviter les situations où une ligne de base de correctifs approuve un correctif alors qu'une autre le bloque, ce qui peut entraîner des problèmes entre les versions conflictuelles.

La stratégie suivante vous permet d'utiliser le paramètre `InstallOverrideList` pour appliquer différents types de correctifs à un groupe cible, selon des calendriers différents, tout en utilisant la même ligne de base de correctifs.

1. Dans la ligne de base du correctif par défaut, assurez-vous que seules les mises à jour `Security` sont spécifiées.
2. Créez une première fenêtre de maintenance qui exécute `AWS-RunPatchBaseline` chaque semaine. Ne spécifiez pas de liste de remplacement.
3. Créez une liste de remplacement des correctifs de tous les types que vous souhaitez appliquer chaque mois et stockez-la dans un compartiment Amazon S3. Un exemple de script que vous pouvez modifier pour créer la liste des correctifs suit cette procédure.

4. Créez une deuxième fenêtre de maintenance qui s'exécute une fois par mois. Mais pour la tâche Fonctionnalité Exécuter la commande que vous inscrivez pour cette fenêtre de maintenance, spécifiez l'emplacement de votre liste de remplacement.

Résultat : Seuls les `Security` correctifs, tels que définis dans votre référence de correctifs par défaut, sont installés chaque semaine. Tous les correctifs disponibles, ou le sous-ensemble de correctifs que vous définissez, sont installés chaque mois.

À propos des références de correctifs

Les rubriques de cette section fournissent des informations sur le fonctionnement des références de correctifs.

Les informations contenues dans les rubriques suivantes s'appliquent lorsque vous appliquez des correctifs à des instances selon une planification et à la demande.

Rubriques

- À propos des références de correctifs prédéfinies et personnalisées (p. 1098)
- À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés (p. 1103)
- À propos des groupes de correctifs (p. 1105)
- À propos des correctifs d'applications sur Windows Server (p. 1108)

À propos des références de correctifs prédéfinies et personnalisées

Une référence de correctifs définit les correctifs qui sont approuvés pour l'installation sur vos instances. Vous pouvez spécifier un par un les correctifs approuvés ou rejetés. Vous pouvez également utiliser les règles d'approbation automatique pour spécifier que certains types de mises à jour (par exemple, les mises à jour critiques) doivent être approuvés automatiquement. La liste des mises à jour rejetées remplace à la fois les règles et la liste des mises à jour approuvées.

Pour utiliser une liste de correctifs approuvés pour installer des packages spécifiques, commencez par supprimer toutes les règles d'approbation automatique. Si vous avez identifié explicitement un correctif en tant que rejeté, il ne sera ni approuvé, ni installé, même s'il correspond à tous les critères d'une règle d'approbation automatique. De la même façon, un correctif est installé sur une instance à condition qu'il s'applique au logiciel sur l'instance, même si le correctif a été approuvé pour l'instance par ailleurs.

Gestionnaire de correctifsfournit des références de correctifs prédéfinies pour chacun des systèmes d'exploitation pris en charge par Gestionnaire de correctifs . Vous pouvez utiliser ces lignes de base telles qu'elles sont actuellement configurées (vous ne pouvez pas les personnaliser) ou vous pouvez créer vos propres lignes de base de correctifs personnalisées. Les références de correctifs personnalisées vous permettent de mieux contrôler les correctifs approuvés ou rejetés pour votre environnement. En outre, les lignes de base prédéfinies attribuent un niveau de conformité `Unspecified` à tous les correctifs installés à l'aide de ces lignes de base. Pour que les valeurs de conformité soient affectées, vous pouvez créer une copie d'une ligne de base prédéfinie et spécifier les valeurs de conformité que vous souhaitez affecter aux correctifs. Pour de plus amples informations, veuillez consulter À propos des références personnalisées (p. 1101) et Utilisation de références de correctifs personnalisées (p. 1127).

Rubriques

- À propos des références prédéfinies (p. 1099)
- À propos des références personnalisées (p. 1101)

À propos des références prédéfinies

Le tableau suivant décrit les références de correctifs prédéfinies fournies avec Gestionnaire de correctifs .

Pour plus d'informations sur les versions de chaque système d'exploitation Gestionnaire de correctifs prises en charge, consultez Prérequis d'Gestionnaire de correctifs (p. 1048) .

Nom	Système d'exploitation pris en charge	Details (Détails)
`AWS-AmazonLinuxDefaultPatchBaseline`	Amazon Linux	Approuve tous les correctifs de système d'exploitation qui sont classés dans la section « Security (Sécurité) » et qui ont un niveau de gravité « Critical (Critique) » ou « Important ». Les correctifs sont approuvés automatiquement sept jours après leur publication. Approuve également automatiquement tous les correctifs avec une classification « Bugfix » sept jours après leur publication
`AWS-AmazonLinux2DefaultPatchBaseline`	Amazon Linux 2	Approuve tous les correctifs de système d'exploitation qui sont classés dans la section « Security (Sécurité) » et qui ont un niveau de gravité « Critical (Critique) » ou « Important ». Les correctifs sont approuvés automatiquement sept jours après leur publication. Approuve également tous les correctifs avec une classification « Bugfix (correctif de bogue) » sept jours après leur publication.
`AWS-CentOSDefaultPatchBaseline`	CentOS	Approuve tous les mises à jour 7 jours après leur mise à disponibilité, y compris les mises à jour non liées à la sécurité.
`AWS-DebianDefaultPatchBaseline`	Debian Server	Approuve immédiatement tous les correctifs de système d'exploitation relatifs à la sécurité qui ont une priorité « Required (Obligatoire) », « Important (Importante) », « Standard », « Optional (Facultative) » ou « Extra ». L'approbation est immédiate, car aucune date de publication fiable n'est disponible dans le référentiel.
`AWS-OracleLinuxDefaultPatchBaseline`	Oracle Linux	Approuve tous les correctifs de système d'exploitation qui sont classés dans la section

Nom	Système d'exploitation pris en charge	Details (Détails)
		« Security (Sécurité) » et qui ont un niveau de gravité « Important » ou « Moderate (Modéré) ». Les correctifs sont approuvés automatiquement sept jours après leur publication. Approuve également tous les correctifs qui sont classés en tant que « Bugfix (Correctif de bogue) » sept jours après leur publication.
`AWS-RedHatDefaultPatchBaseline`	Red Hat Enterprise Linux (RHEL)	Approuve tous les correctifs de système d'exploitation qui sont classés dans la section « Security (Sécurité) » et qui ont un niveau de gravité « Critical (Critique) » ou « Important ». Les correctifs sont approuvés automatiquement sept jours après leur publication. Approuve également tous les correctifs qui sont classés en tant que « Bugfix (Correctif de bogue) » sept jours après leur publication.
`AWS-SuseDefaultPatchBaseline`	SUSE Linux Enterprise Server (SLES)	Approuve tous les correctifs de système d'exploitation qui sont classés en tant que « Security (Sécurité) » et qui ont un niveau de gravité « Critical (Critique) » ou « Important ». Les correctifs sont approuvés automatiquement sept jours après leur publication.
`AWS-UbuntuDefaultPatchBaseline`	Ubuntu Server	Approuve immédiatement tous les correctifs de système d'exploitation relatifs à la sécurité qui ont une priorité « Required (Obligatoire) », « Important (Importante) », « Standard », « Optional (Facultative) » ou « Extra ». L'approbation est immédiate, car aucune date de publication fiable n'est disponible dans le référentiel.

Nom	Système d'exploitation pris en charge	Details (Détails)
`AWS-DefaultPatchBaseline`	Windows Server	Approuve tous les correctifs de système d'exploitation Windows Server qui sont classés en tant que « CriticalUpdates (Mises à jour critiques) » ou « SecurityUpdates (Mises à jour de sécurité) » et qui ont un niveau de gravité MSRC « Critical (Critique) » ou « Important ». Les correctifs sont approuvés automatiquement sept jours après leur publication.
`AWS-WindowsPredefinedPatchBaseline-OS`	Windows Server	Approuve tous les correctifs de système d'exploitation Windows Server qui sont classés en tant que « CriticalUpdates (Mises à jour critiques) » ou « SecurityUpdates (Mises à jour de sécurité) » et qui ont un niveau de gravité MSRC « Critical (Critique) » ou « Important ». Les correctifs sont approuvés automatiquement sept jours après leur publication.
`AWS-WindowsPredefinedPatchBaseline-OS-Applications`	Windows Server	Pour le système d'exploitation Windows Server, approuve tous les correctifs qui sont classés en tant que « CriticalUpdates (Mises à jour critiques) » ou « SecurityUpdates (Mises à jour de sécurité) » et qui ont un niveau de gravité MSRC « Critical (Critique) » ou « Important ». Pour les applications Microsoft, approuve tous les correctifs. Les correctifs de système d'exploitation et d'applications sont automatiquement approuvés sept jours après leur publication.

À propos des références personnalisées

Si vous créez votre propre référence de correctifs, vous pouvez choisir les correctifs à approuver automatiquement en utilisant les catégories suivantes.

- Système d'exploitation : Windows, Amazon Linux , , etc.Ubuntu Server
- Nom du produit (pour les systèmes d'exploitation) : Par exemple, RHEL 6.5, Amazon Linux 2014.09, Windows Server 2012, Windows Server 2012 R2, etc.
- Nom du produit (pour les applications Microsoft sur Windows Server uniquement) : Par exemple, Word 2016, BizTalk Server, etc.

- Classification : Par exemple, les mises à jour critiques, les mises à jour de sécurité, etc.
- Gravité : Par exemple, critique, important, etc.

Pour chaque règle d'approbation que vous créez, vous pouvez choisir de spécifier un délai d'approbation automatique ou une date limite d'approbation des correctifs.

> **Note**
>
> Comme il n'est pas possible de déterminer de manière fiable les dates de publication des packages de mise à jour pour Ubuntu Server, les options d'approbation automatique ne sont pas prises en charge pour ce système d'exploitation.

Un délai d'auto-approbation correspond au nombre de jours à attendre après la publication du correctif, avant l'approbation automatique de l'application de celui-ci. Par exemple, si vous créez une règle à l'aide de la classification `CriticalUpdates` et que vous la configurez pour un délai de sept jours avant l'approbation automatique, un nouveau patch critique publié le 7 juillet sera automatiquement approuvé le 14 juillet.

> **Note**
>
> Si un référentiel Linux ne fournit pas les informations relatives à la date de publication des packages, Systems Manager utilise l'heure de génération du package comme délai d'approbation automatique pour Amazon Linux , Amazon Linux 2 , et RHEL .CentOS Si le système ne parvient pas à trouver le temps de génération du package, Systems Manager traite le délai d'approbation automatique comme ayant une valeur de zéro.

Lorsque vous spécifiez une date limite d'approbation automatique, applique Gestionnaire de correctifs automatiquement tous les correctifs publiés à cette date ou avant cette date. Par exemple, si vous spécifiez le 7 juillet 2020 comme date limite, aucun correctif publié à partir du 8 juillet 2020 n'est installé automatiquement.

Vous pouvez également spécifier un niveau de sécurité de conformité. Si un correctif approuvé est signalé comme manquant, `Compliance Level` est la gravité de violation de conformité.

Si vous utilisez plusieurs références de correctifs avec des délais d'approbation automatique ou des dates limites différents, vous pouvez déployer des correctifs à des rythmes différents pour différentes instances. Par exemple, vous pouvez créer des références de correctifs, des délais d'approbation automatique et des dates limites distincts pour les environnements de développement et de production. Cela vous permet de tester les correctifs dans votre environnement de développement avant leur déploiement dans votre environnement de production.

Gardez les points suivants à l'esprit lorsque vous créez une référence de correctifs :

- Gestionnaire de correctifsfournit une référence de correctifs prédéfinie pour chaque système d'exploitation pris en charge. Ces références de correctifs prédéfinies sont utilisées en tant que références de correctifs par défaut pour chaque type de système d'exploitation, sauf si vous créez votre propre référence de correctifs et que vous la définissez comme la référence par défaut pour le type de système d'exploitation correspondant.

  > **Note**
  >
  > Pour Windows Server, deux références de correctifs prédéfinies sont fournies. La référence de correctifs `AWS-WindowsPredefinedPatchBaseline-OS` prend en charge uniquement les mises à jour du système d'exploitation sur le système d'exploitation Windows lui-même. Elle est utilisée comme référence de correctifs par défaut pour les instances Windows Server, sauf si vous spécifiez une autre référence de correctifs. L'autre référence de correctifs Windows prédéfinie, `AWS-WindowsPredefinedPatchBaseline-OS-Applications`, peut être utilisée pour appliquer des correctifs au système d'exploitation Windows Server et aux applications Microsoft prises en charge.

- Pour les instances sur site ou autres instances non-EC2, Gestionnaire de correctifs tente d'utiliser votre référence de correctifs personnalisée par défaut. S'il n'existe aucune référence de correctifs personnalisée par défaut, le système utilise la référence de correctifs prédéfinie pour le système d'exploitation correspondant.
- Si un correctif est répertorié comme approuvé et refusé dans la même référence de correctifs, le correctif est refusé.
- Une instance peut uniquement contenir une référence de correctifs définie.
- Les formats de noms de package que vous pouvez ajouter à la liste des correctifs approuvés et rejetés pour une référence de correctifs dépendent du type de système d'exploitation auquel vous appliquez des correctifs.

  Pour plus d'informations sur les formats acceptés pour les listes de correctifs approuvés et de correctifs rejetés, consultez À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés (p. 1103).

Pour plus d'informations sur la création d'une référence de correctifs, consultez Utilisation de références de correctifs personnalisées (p. 1127) et Procédure : Appliquer un correctif à un environnement serveur (AWS CLI) (p. 1168).

À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés

Les formats de noms de package que vous pouvez ajouter à la liste des correctifs approuvés et rejetés dépendent du type de système d'exploitation auquel vous appliquez des correctifs.

Formats de noms de package pour les systèmes d'exploitation Linux

Les formats que vous pouvez spécifier pour les correctifs approuvés et rejetés dans votre référence de correctifs varient selon le type de système Linux. Plus précisément, les formats qui sont pris en charge dépendent du gestionnaire de package utilisé par le type de système d'exploitation Linux.

Rubriques

- Amazon Linux, Amazon Linux 2 , CentOS , et Oracle Linux (Red Hat Enterprise Linux)RHEL (p. 1103)
- Debian Server et Ubuntu Server (p. 1104)
- SUSE Linux Enterprise Server (SLES) (p. 1104)

Amazon Linux, Amazon Linux 2 , CentOS , et Oracle Linux (Red Hat Enterprise Linux)RHEL

Gestionnaire de package : YUM, à l'exception de RHEL 8 et CentOS 8, qui utilisent DNF comme gestionnaire de package

Correctifs approuvés : Pour les correctifs approuvés, vous pouvez spécifier l'un des éléments suivants :

- Bugzilla, dans le format IDs (le système traite les chaînes composées uniquement de chiffres en tant que Bugzilla `1234567` .)IDs
- CVE, au formatIDs`CVE-2018-1234567`
- Advisory, dans des formats tels que IDs et`RHSA-2017:0864ALAS-2018-123`
- Noms de package complets, dans des formats tels que :
  - `example-pkg-0.710.10-2.7.abcd.x86_64`
  - `pkg-example-EE-20180914-2.2.amzn1.noarch`
- Noms de package avec un seul caractère générique, dans des formats tels que :
  - `example-pkg-*.abcd.x86_64`
  - `example-pkg-*-20180914-2.2.amzn1.noarch`

- `example-pkg-EE-2018*.amzn1.noarch`

Correctifs rejetés : Pour les correctifs rejetés, vous pouvez spécifier l'un des éléments suivants :

- Noms de package complets, dans des formats tels que :
  - `example-pkg-0.710.10-2.7.abcd.x86_64`
  - `pkg-example-EE-20180914-2.2.amzn1.noarch`
- Noms de package avec un seul caractère générique, dans des formats tels que :
  - `example-pkg-*.abcd.x86_64`
  - `example-pkg-*-20180914-2.2.amzn1.noarch`
  - `example-pkg-EE-2018*.amzn1.noarch`

Debian Server et Ubuntu Server

Gestionnaire de package : APT

Correctifs approuvés et rejetés : Pour les correctifs approuvés et rejetés, spécifiez les informations suivantes :

- Noms de package, au format `ExamplePkg33`

  Note

  Pour les listes Ubuntu Server, n'incluez pas les éléments tels que l'architecture ou les versions. Par exemple, vous spécifiez le nom de package `ExamplePkg33` pour inclure tous les éléments suivants dans une liste de correctifs :

  - `ExamplePkg33.x86.1`
  - `ExamplePkg33.x86.2`
  - `ExamplePkg33.x64.1`
  - `ExamplePkg33.3.2.5-364.noarch`

SUSE Linux Enterprise Server (SLES)

Gestionnaire de package : Zypper

Correctifs approuvés et rejetés : Pour les listes de correctifs approuvés et rejetés, vous pouvez spécifier l'un des éléments suivants :

- Noms de package complets, dans des formats tels que :
  - `SUSE-SLE-Example-Package-12-2018-123`
  - `example-pkg-2018.11.4-46.17.1.x86_64.rpm`
- Noms de package avec un seul caractère générique, tels que :
  - `SUSE-SLE-Example-Package-12-2018-*`
  - `example-pkg-2018.11.4-46.17.1.*.rpm`

Formats de noms de package pour les systèmes d'exploitation Windows

Pour les systèmes d'exploitation Windows, spécifiez les correctifs à l'aide de la base de connaissances Microsoft IDs et du bulletin de sécurité Microsoft IDs ; par exemple :

```
KB2032276,KB2124261,MS10-048
```

À propos des groupes de correctifs

Vous pouvez utiliser un groupe de correctifs pour associer des instances à une référence de correctifs spécifique. Les groupes de correctifs vous permettent de vous assurer que vous déployez les correctifs appropriés, en fonction des règles de référence de correctifs connexes, pour l'ensemble d'instances adéquat. Les groupes de correctifs peuvent également vous aider à éviter le déploiement de correctifs avant qu'ils aient été testés correctement. Par exemple, vous pouvez créer des groupes de correctifs pour différents environnements (par exemple, développement, test ou production) et enregistrer chaque groupe de correctifs dans une référence de correctifs appropriée.

> **Note**
>
> Un groupe de correctifs peut être enregistré avec une seule référence de correctifs pour chaque type de système d'exploitation.

Lorsque vous exécutez `AWS-RunPatchBaseline` , vous pouvez cibler les instances gérées à l'aide de leur ID d'instance ou de leurs balises, Agent SSM puis évaluer quelle référence de correctifs utiliser en fonction de la valeur du groupe de correctifs que vous avez ajoutée à l'instance.Gestionnaire de correctifs

Vous créez un groupe de correctifs à l'aide de balises Amazon EC2. Contrairement aux autres scénarios de balisage sur Systems Manager , un groupe de correctifs doit être défini avec la clé de balise : Groupe de correctifs. Notez que la clé est sensible à la casse. Vous pouvez spécifier une valeur, par exemple « serveurs web », mais la clé doit être Groupe de correctifs.

> **Note**
>
> Une instance peut appartenir à un seul groupe de correctifs.

Une fois que vous avez créé un groupe de correctifs et des instances de balise, vous pouvez enregistrer le groupe de correctifs dans une référence de correctifs. L'enregistrement du groupe de correctifs dans une référence de correctifs garantit que les instances au sein du groupe de correctifs utilisent les règles définies dans la référence de correctifs associée. Pour de plus amples informations sur la création d'un groupe de correctifs et sur l'association de ce groupe à une référence de correctifs, veuillez consulter Création d'un groupe de correctifs (p. 1137) et Ajout d'un groupe de correctifs à une référence de correctifs (p. 1138).

Pour voir un exemple de création d'une référence de correctifs et de groupes de correctifs à l'aide de l'AWS CLI, consultez Procédure : Appliquer un correctif à un environnement serveur (AWS CLI) (p. 1168). Pour de plus amples informations sur les balises Amazon EC2, veuillez consulter Balisage de vos ressources Amazon EC2 dans le Guide de l'utilisateur Amazon EC2.

Fonctionnement

Lorsque le système exécute la tâche d'application d'une référence de correctifs à une instance, l'agent SSM vérifie si une valeur de groupe de correctifs est définie pour l'instance. Si l'instance est affectée à un groupe de correctifs, vérifie quelle référence de correctifs est enregistrée dans ce groupe.Gestionnaire de correctifs Si une référence de correctifs est trouvée pour ce groupe, informe Gestionnaire de correctifs d'utiliser la référence de correctifs associée.Agent SSM Si une instance n'est pas configurée pour un groupe de correctifs, demande Gestionnaire de correctifs automatiquement Agent SSM à d'utiliser la référence de correctifs par défaut actuellement configurée.

Le schéma suivant illustre un exemple général des processus Systems Manager exécutés par lors de l'envoi d'une Fonctionnalité Exécuter la commande tâche à votre parc de serveurs pour appliquer des correctifs à l'aide d' Gestionnaire de correctifs . Un processus similaire est utilisé lorsqu'une fenêtre de maintenance est configurée pour envoyer une commande d'application de correctifs à l'aide d' Gestionnaire de correctifs .

Cet exemple illustre trois groupes d'instances EC2 pour Windows Server avec les balises suivantes appliquées :

Groupe d'instances EC2	Tags
Groupe 1	`key=OS,value=Windows` `key=Patch Group,value=DEV`
Groupe 2	`key=OS,value=Windows`
Groupe 3	`key=OS,value=Windows` `key=Patch Group,value=QA`

Dans cet exemple, nous avons également ces deux références de correctifs Windows :

ID de référence de correctif	Valeur par défaut	Groupe de correctifs associé
`pb-0123456789abcdef0`	Oui	`Default`
`pb-9876543210abcdef0`	Non	`DEV`

Diagramme 1 : Exemple général de flux de processus d'opérations d'application de correctifs

Le processus général d'analyse ou d'installation des correctifs à l'aide Fonctionnalité Exécuter la commande d' Gestionnaire de correctifs se présente comme suit :

1. Envoyez une commande pour corriger : Utilisez la Systems Manager console , le kit SDK AWS CLI , lOutils AWS pour Windows PowerShell' ou l' pour envoyer une tâche Exécuter la commande à l'aide du document `AWS-RunPatchBaseline` . Le schéma montre une tâche Exécuter la commande pour corriger les instances gérées en ciblant la balise `key=OS,value=Windows` .
2. Détermination de la référence de correctifs : Agent SSM vérifie les balises de groupe de correctifs appliquées à l'instance EC2 et les requêtes Gestionnaire de correctifs pour la référence de correctifs correspondante.
   - Mise en correspondance de la valeur du groupe de correctifs associée avec la référence de correctifs :
     1. Agent SSM, qui est installé sur les instances EC2 dans le groupe un, reçoit la commande émise à l'étape 1 pour commencer une opération d'application de correctifs. Agent SSM valide que

la valeur de balise du groupe de correctifs `DEV` appliquée aux instances EC2 et les requêtes Gestionnaire de correctifs pour une référence de correctifs associée.

2. Gestionnaire de correctifsvérifie que le groupe de correctifs `pb-9876543210abcdef0` est associé à la référence de correctifs et en informe `DEV` .Agent SSM
3. Agent SSMextrait un instantané de référence de correctifs à partir d' Gestionnaire de correctifs en fonction des règles d'approbation et des exceptions configurées dans `pb-9876543210abcdef0` et passe à l'étape suivante.

- Aucune balise de groupe de correctifs n'est ajoutée à l'instance :

1. Agent SSM, qui est installé sur les instances EC2 du groupe 2, reçoit la commande émise à l'étape 1 pour commencer une opération d'application de correctifs. Agent SSM valide que les instances EC2 n'ont pas de `Patch Group` balise appliquée et, par conséquent, Agent SSM interroge Gestionnaire de correctifs la référence de correctifs Windows par défaut.
2. Gestionnaire de correctifsvérifie que la référence de Windows Server correctifs par défaut est `pb-0123456789abcdef0` et en informe Agent SSM .
3. extrait un instantané de référence de correctifs à partir d' Agent SSM en fonction des règles d'approbation et des exceptions configurées dans la référence de correctifs par défautGestionnaire de correctifs, puis passe à l'étape suivante.`pb-0123456789abcdef0`

- Aucune valeur de groupe de correctifs correspondante n'est associée à une référence de correctifs :

1. Agent SSM, qui est installé sur les instances EC2 du groupe 3, reçoit la commande émise à l'étape 1 pour commencer une opération d'application de correctifs. Agent SSM valide que la valeur de balise du groupe de correctifs `QA` appliquée aux instances EC2 et les requêtes Gestionnaire de correctifs pour une référence de correctifs associée.
2. Gestionnaire de correctifsne trouve pas de référence de correctifs `QA` associée au groupe de correctifs.
3. Gestionnaire de correctifsinforme Agent SSM d'utiliser la référence de correctifs Windows par défaut`pb-0123456789abcdef0`.
4. Agent SSMextrait un instantané de référence de correctifs à partir d' Gestionnaire de correctifs en fonction des règles d'approbation et des exceptions configurées dans la référence de correctifs par défaut`pb-0123456789abcdef0`, puis passe à l'étape suivante.

3. Analyse ou installation de correctifs : Après avoir déterminé la référence de correctifs appropriée à utiliser, Agent SSM commence à rechercher ou à installer les correctifs en fonction de la valeur d'opération spécifiée à l'étape 1. Les correctifs qui sont recherchés ou installés sont déterminés par les règles d'approbation et les exceptions de correctifs définies dans l'instantané de référence de correctifs fourni par Gestionnaire de correctifs .

Contenu connexe

- Présentation des valeurs d'état de conformité des correctifs (p. 1113)

À propos des correctifs d'applications sur Windows Server

Pour Windows Server, deux références de correctifs prédéfinies sont fournies. La référence de correctifs `AWS-WindowsPredefinedPatchBaseline-OS` prend en charge uniquement les mises à jour du système d'exploitation sur le système d'exploitation Windows lui-même. Elle est utilisée comme référence de correctifs par défaut pour les instances Windows Server, sauf si vous spécifiez une autre référence de correctifs. L'autre référence de correctifs Windows prédéfinie, `AWS-WindowsPredefinedPatchBaseline-OS-Applications`, peut être utilisée pour appliquer des correctifs au système d'exploitation Windows Server et aux applications Microsoft prises en charge.

> **Note**
>
> L'application des correctifs aux applications Microsoft est disponible uniquement sur les instances EC2 et dans le niveau d'instances avancées. Pour corriger les applications Microsoft sur les serveurs sur site et VMs , vous devez activer le niveau d'instances avancées. Pour plus d'informations, consultez la section Activation du niveau d'instances avancées (p. 853).

Vous pouvez également créer une référence de correctifs personnalisée pour mettre à jour les applications Microsoft sur les machines Windows Server.

Pour inclure des applications Microsoft dans votre référence de correctifs personnalisée, vous devez, au minimum, spécifier le produit auquel vous souhaitez appliquer un correctif. La commande de l'AWS CLI suivante illustre les exigences minimales pour appliquer un correctif à un produit, comme Office 2016 :

Linux

```
aws ssm create-patch-baseline \
    --name "My-Windows-App-Baseline" \
    --approval-rules
 "PatchRules=[{PatchFilterGroup={PatchFilters=[{Key=PRODUCT,Values='Office 2016'},
{Key=PATCH_SET,Values='APPLICATION'}]},ApproveAfterDays=5}]"
```

Windows

```
aws ssm create-patch-baseline ^
    --name "My-Windows-App-Baseline" ^
    --approval-rules
 "PatchRules=[{PatchFilterGroup={PatchFilters=[{Key=PRODUCT,Values='Office 2016'},
{Key=PATCH_SET,Values='APPLICATION'}]},ApproveAfterDays=5}]"
```

Si vous spécifiez la famille de produits des applications Microsoft, chaque produit que vous spécifiez doit être un membre pris en charge de la famille de produits sélectionnée. Par exemple, pour appliquer un correctif au produit « Active Directory Rights Management Services Client 2.0 », vous devez spécifier sa famille de produits sous la forme « Active Directory » et non pas, par exemple, sous la forme « Office » ou « SQL Server ». La commande de l'AWS CLI suivante illustre un appairage de familles de produits et de produits :

Linux

```
aws ssm create-patch-baseline \
    --name "My-Windows-App-Baseline" \
    --approval-rules
 "PatchRules=[{PatchFilterGroup={PatchFilters=[{Key=PRODUCT_FAMILY,Values='Active
 Directory'},{Key=PRODUCT,Values='Active Directory Rights Management Services Client
 2.0'},{Key=PATCH_SET,Values='APPLICATION'}]},ApproveAfterDays=5}]"
```

Windows

```
aws ssm create-patch-baseline ^
    --name "My-Windows-App-Baseline" ^
    --approval-rules
 "PatchRules=[{PatchFilterGroup={PatchFilters=[{Key=PRODUCT_FAMILY,Values='Active
 Directory'},{Key=PRODUCT,Values='Active Directory Rights Management Services Client
 2.0'},{Key=PATCH_SET,Values='APPLICATION'}]},ApproveAfterDays=5}]"
```

Résolution des problèmes liés aux paires familles de produits/produits non assorties

Lorsque vous créez une référence de correctifs dans la console, vous spécifiez une famille de produits et un produit. Par exemple, vous pouvez choisir ce qui suit :

- Famille de produits : **`Office`**

  Produit :**`Office 2016`**

Si vous essayez de créer une référence de correctifs avec une paire famille de produits/produit non assortie, un message d'erreur s'affiche. Voici les cas où cette situation peut se présenter :

- Vous avez sélectionné une paire famille de produits/produit valide, puis supprimé la sélection de la famille de produits.
- Vous avez choisi un produit dans la sous-liste Obsolete or mismatched options (Options obsolètes ou non assorties) au lieu de la sous-liste Available and matching options (Options disponibles et assorties).

  Les éléments de la sous-liste de produits Obsolete or mismatched options (Options obsolètes ou non assorties) peuvent avoir été saisis de façon erronée via un kit SDK ou une commande de l'AWS CLI `create-patch-baseline`. Cela peut signifier qu'une faute de frappe a été introduite ou qu'un produit a été attribué à la mauvaise famille de produits. Un produit apparaît également dans la sous-liste Obsolete or mismatched options (Options obsolètes ou non assorties) s'il a été spécifié pour une référence de correctifs précédente mais qu'aucun correctif n'est actuellement disponible à partir de Microsoft.

  Pour éviter ce problème dans la console, choisissez toujours des options des sous-listes Currently available options (Options actuellement disponibles).

  Vous pouvez également consulter les produits pour lesquels des correctifs sont actuellement disponibles à l'aide de la commande `describe-patch-properties` dans l'AWS CLI ou de la commande d'API `DescribePatchProperties`.

Correction d'instances hors conformité (Gestionnaire de correctifs)

Les rubriques de cette section fournissent des aperçus sur la façon d'identifier les instances gérées qui ne sont pas conformes aux correctifs et la façon de les rendre conformes.

Rubriques

- Identification des instances hors conformité (p. 1110)
- Présentation des valeurs d'état de conformité des correctifs (p. 1113)
- Application de correctifs aux instances non conformes (p. 1117)

Identification des instances hors conformité

Les instances hors conformité sont identifiées lorsque l'un des deux Systems Manager documents (documents SSM) est exécuté. Ces documents SSM référencent la référence de correctifs appropriée pour chaque instance, évaluent l'état des correctifs de l'instance, puis mettent les résultats de conformité à votre disposition.

Deux documents SSM sont utilisés pour identifier ou mettre à jour les instances hors conformité : `AWS-RunPatchBaseline`et `AWS-RunPatchBaselineAssociation` . Chacun est utilisé par différents processus, et leurs résultats de conformité sont disponibles via différents canaux. Le tableau suivant décrit les différences entre ces documents.

Note

Les données de conformité des correctifs du Gestionnaire de correctifs peuvent être envoyées à AWS Security Hub. Security Hub vous offre une vue complète des alertes de sécurité hautement prioritaires et de votre statut de conformité. Il surveille également le statut de l'application des correctifs de votre flotte. Pour de plus amples informations, veuillez consulter Intégration d'Gestionnaire de correctifs à AWS Security Hub (p. 1145).

	`AWS-RunPatchBaseline`	**`AWS-RunPatchBaselineAssociation`**
Processus qui utilisent le document	Correctif à la demande – Vous pouvez analyser ou corriger les instances à la demande à l'aide de l'option Patch now. Pour plus d'informations, consultez la section Application de correctifs à des instances à la demande (p. 1139). Configuration d'Gestionnaire de correctifsapplication de correctifs – : vous pouvez créer une configuration d'application de correctifs qui inclut une fenêtre de maintenance pour analyser les instances de conformité des correctifs sur un calendrier. Pour plus d'informations, consultez la section Création d'une configuration d'application de correctifs (console) (p. 1135). Exécuter une commande – : vous pouvez exécuter manuellement `AWS-RunPatchBaseline` dans une Fonctionnalité Exécuter la commande opération. Pour plus d'informations, consultez la section Exécution des commandes depuis la console (p. 946). Fenêtre de maintenance – : vous pouvez créer une fenêtre de maintenance qui utilise le Systems Manager document (document SSM) `AWS-RunPatchBaseline` dans un type Fonctionnalité Exécuter la commande de tâche. Pour plus d'informations, consultez la section Création d'une fenêtre de maintenance pour l'application des correctifs (p. 1141).	Systems ManagerConfiguration rapide (p. 17) – : vous pouvez configurer la configuration rapide pour utiliser Gestionnaire de correctifs afin d'analyser chaque jour la conformité des correctifs de vos instances gérées. Pour plus d'informations, consultez Analyser quotidiennement les instances pour rechercher les correctifs manquants (p. 22) dans la rubrique AWS Systems Manager Configuration rapide (p. 17) . Systems ManagerExplorer (p. 153)– Lorsque vous activez Explorer , il analyse régulièrement vos instances gérées pour vérifier la conformité des correctifs et signale les résultats dans le Explorer tableau de bord.

	AWS-RunPatchBaseline	**AWS-RunPatchBaselineAssociation**
Format des données de résultat de l'analyse des correctifs	Après les `AWS-RunPatchBaseline` exécutions, Gestionnaire de correctifs envoie un `AWS:PatchSummary` objet à Systems Manager l'inventaire .	Après les `AWS-RunPatchBaselineAssociation` exécutions, Gestionnaire de correctifs envoie un `AWS:ComplianceItem` objet à Systems Manager l'inventaire .
Affichage des rapports de conformité des correctifs dans la console	Vous pouvez afficher les informations de conformité des correctifs pour les processus qui utilisent `AWS-RunPatchBaseline` dans Systems Manager Conformité de configuration (p. 778) et Instances gérées AWS Systems Manager (p. 851) . Pour plus d'informations, consultez la section Affichage des résultats de conformité des correctifs (p. 1125).	Si vous utilisez la configuration rapide pour analyser vos instances gérées afin de vérifier la conformité des correctifs, vous pouvez consulter le rapport de conformité dans Systems Manager State Manager (p. 987), qui est accessible à l'aide d'un bouton View results (Afficher les résultats) dans la configuration rapide. Si vous utilisez Explorer pour analyser vos instances gérées afin de vérifier la conformité des correctifs, vous pouvez consulter le rapport de conformité dans Explorer et Systems Manager OpsCenter (p. 176) .
AWS CLICommandes pour l'affichage des résultats de conformité des correctifs	Pour les processus qui utilisent `AWS-RunPatchBaseline` , vous pouvez utiliser les commandes CLI suivantes pour afficher des informations récapitulatives sur les correctifs d'une instance. • describe-instance-patch-states • describe-instance-patch-states-for-patch-group • describe-patch-group-state	Pour les processus qui utilisent `AWS-RunPatchBaselineAssociation` , vous pouvez utiliser la commande suivante de l'interface de ligne de commande pour afficher des informations récapitulatives sur les correctifs d'une instance. • list-compliance-items
Opérations d'application de correctifs	Pour les processus qui utilisent `AWS-RunPatchBaseline` , vous spécifiez si vous souhaitez que l'opération exécute uniquement une `Scan` opération ou une `Scan and install` opération . Si votre objectif est d'identifier les instances hors conformité et de ne pas les corriger, exécutez uniquement une `Scan` opération .	Les Explorer processus et la configuration rapide, qui utilisent `AWS-RunPatchBaselineAssociation` , exécutent uniquement une `Scan` opération .

	AWS-RunPatchBaseline	**AWS-RunPatchBaselineAssociation**
En savoir plus	À propos du document SSM AWS-RunPatchBaselineAssociation (p. 1088)	À propos du document SSM AWS-RunPatchBaselineAssociation (p. 1088)

Pour plus d'informations sur les différents états de conformité des correctifs signalés, consultez .Présentation des valeurs d'état de conformité des correctifs (p. 1113)

Pour plus d'informations sur la résolution des instances gérées non conformes aux correctifs, consultez Application de correctifs aux instances non conformes (p. 1117) .

Présentation des valeurs d'état de conformité des correctifs

Les informations sur les correctifs pour une instance incluent un rapport de l'état, ou du statut, de chaque correctif individuel.

> **Note**
>
> Si vous souhaitez attribuer un état de conformité des correctifs spécifique à une instance, vous pouvez utiliser la commande de l'put-compliance-itemsinterface de ligne de commande ou l'action PutComplianceItems d'API. L'attribution d'un état de conformité n'est pas prise en charge dans la console.

Aidez-vous des informations des tableaux suivants pour vous aider à identifier pourquoi une instance peut ne pas être conforme aux correctifs.

Valeurs de conformité des correctifs pour Debian Server etUbuntu Server

Pour Debian Server et Ubuntu Server , les règles de classification des packages dans les différents états de conformité sont décrites dans le tableau suivant.

> **Note**
>
> Gardez les points suivants à l'esprit lorsque vous évaluez les valeurs de statut Installed, Installed Other et Missing : Si vous ne cochez pas la case Inclure les mises à jour non liées à la sécurité lors de la création ou de la mise à jour d'une référence de correctifs, les versions possibles des correctifs sont limitées aux correctifs inclus dans `trusty-security` (Ubuntu Server 14), `xenial-security` (Ubuntu Server 16), (`bionic-security` 18), Ubuntu Server (`focal-security` 20) ou Ubuntu Server (`debian-security`Debian Stretch ou Jessie. Si vous cochez la case Include nonsecurity updates (Inclure les mises à jour non liées à la sécurité), les correctifs d'autres référentiels sont également pris en compte.

État du correctif	Description	Statut de conformité
Installé	Le correctif est répertorié dans la référence de correctif et est installé sur l'instance. Elle peut avoir été installée manuellement par une personne ou automatiquement par Gestionnaire de correctifs lorsque le `AWS-RunPatchBaseline` document a été exécuté sur l'instance.	Conforme

État du correctif	Description	Statut de conformité
Autre installé	Le correctif n'est pas inclus dans la référence ou n'est pas approuvé par la référence, mais il est installé sur l'instance. Le correctif peut avoir été installé manuellement, le package peut être une dépendance obligatoire d'un autre correctif approuvé ou le correctif peut avoir été inclus dans une InstallOverrideList opération. Si vous ne spécifiez pas `Block` en tant qu'action Correctifs rejetés, `Installed_Other` les correctifs incluent également les correctifs installés, mais rejetés.	Conforme
En attente de redémarrage installé	LGestionnaire de correctifs`Install`'opération a appliqué le correctif à l'instance, mais l'instance n'a pas été redémarrée depuis l'application du correctif. (Notez que les correctifs installés en dehors d' ne reçoivent jamais le statut Gestionnaire de correctifs .)`INSTALLED_PENDING_REBOOT` Cela signifie généralement que l'`NoReboot`option a été sélectionnée pour le `RebootOption` paramètre lors de la dernière exécution du document AWS-RunPatchBaseline sur l'instance. Pour plus d'informations, consultez la section Nom du paramètre : `RebootOption` (p. 1087).	Non conforme
Rejeté installé	Le correctif est installé sur l'instance, mais il est spécifié dans une liste Correctifs rejetés. Cela signifie généralement que le correctif a été installé avant d'être ajouté à la liste des correctifs rejetés.	Non conforme
Manquant	Packages qui sont filtrés à l'aide de la référence et qui ne sont pas déjà installés.	Non conforme
Failed (Échec)	Packages dont l'installation a échoué pendant l'opération de correctif.	Non conforme

Valeurs de conformité des correctifs pour les autres systèmes d'exploitation

Pour tous les systèmes d'exploitation autres que Debian Server et Ubuntu Server , les règles de classification des packages dans les différents états de conformité sont décrites dans le tableau suivant.

État du correctif	Description	Valeur de conformité
INSTALLED (INSTALLÉ)	Le correctif est répertorié dans la référence de correctif et est installé sur l'instance. Elle peut avoir été installée manuellement par une personne ou automatiquement par Gestionnaire de correctifs lorsque le document AWS-RunPatchBaseline a été exécuté sur l'instance.	Conforme
INSTALLED_OTHER (OTRE_INSTALLÉ)	Le correctif n'est pas dans la référence, mais il est installé sur l'instance. Le correctif peut avoir été installé manuellement ou le package peut être une dépendance obligatoire d'un autre correctif approuvé. Si vous ne spécifiez pas `Block` comme action Correctifs rejetés, `Installed_Other` les correctifs incluent également les correctifs installés, mais rejetés.	Conforme
INSTALLED_REJECTED	Le correctif est installé sur l'instance, mais il est spécifié dans une liste des correctifs rejetés. Cela signifie généralement que le correctif a été installé avant d'être ajouté à la liste des correctifs rejetés.	Non conforme
REBOOT_EN ATTENTE	LGestionnaire de correctifs`Install`'opération a appliqué le correctif à l'instance (ou un correctif a été appliqué à une Windows Server instance en dehors d' Gestionnaire de correctifs), mais l'instance n'a pas été redémarrée depuis l'application du correctif. (Notez que les correctifs installés en dehors d' ne Gestionnaire de correctifs reçoivent jamais le statut `INSTALLED_PENDING_REBOOT` .) Cela signifie généralement que l'`NoReboot`option a été sélectionnée pour le `RebootOption` paramètre	Non conforme

État du correctif	Description	Valeur de conformité
	lors de la dernière exécution du document AWS-RunPatchBaseline sur l'instance. Pour plus d'informations, consultez la section Nom du paramètre : `RebootOption` (p. 1087).	
MANQUANT	Le correctif est approuvé dans la référence, mais il n'est pas installé sur l'instance. Si vous configurez la tâche de document AWS-RunPatchBaseline pour l'analyse (au lieu de l'installation), le système signale ce statut pour les correctifs qui ont été détectés lors de l'analyse, mais qui n'ont pas été installés.	Non conforme
N'EST PAS_APPLICABLE	Le correctif est approuvé dans la référence, mais le service ou la fonction qui l'utilise n'est pas installé sur l'instance. Par exemple, un correctif pour un service de serveur web tel qu'Internet Information Services (IIS) indique `NOT_APPLICABLE` s'il a été approuvé dans la référence, mais que le service web n'est pas installé sur l'instance. Un patch peut également être marqué `NOT_APPLICABLE` s'il a été remplacé par une mise à jour ultérieure. Cela signifie que la mise à jour ultérieure est installée et que la `NOT_APPLICABLE` mise à jour n'est plus requise. Note Ce statut de conformité est uniquement signalé sur les systèmes d'exploitation Windows Server.	Ne s'applique pas
ÉCHEC	Le correctif est approuvé dans la référence, mais il n'a pas pu être installé. Pour résoudre ce problème, passez en revue la sortie de commande afin d'accéder à des informations supplémentaires susceptibles de vous aider à comprendre ce qui se passe.	Non conforme

Application de correctifs aux instances non conformes

Un grand nombre des mêmes Systems Manager outils et processus que vous pouvez utiliser pour vérifier la conformité des instances de correctifs peuvent être utilisés pour mettre en conformité les instances avec les règles de correctifs qui s'appliquent actuellement à elles. Pour mettre les instances en conformité avec les correctifs, Gestionnaire de correctifs doit exécuter une `Scan and install` opération. (Si votre objectif est uniquement d'identifier les instances hors conformité et non de les corriger, exécutez une `Scan` opération à la place. Pour plus d'informations, consultez Identification des instances hors conformité (p. 1110) .)

Installer des correctifs à l'aide deSystems Manager

Vous pouvez choisir parmi plusieurs outils pour exécuter une `Scan and install` opération :

- Créez une configuration d'application de correctifs qui inclut une fenêtre de maintenance pour installer les correctifs selon une planification. Pour plus d'informations, consultez la section Création d'une configuration d'application de correctifs (console) (p. 1135).
- Crée une fenêtre de maintenance qui utilise le Systems Manager document (document SSM) `AWS-RunPatchBaseline` dans un type Fonctionnalité Exécuter la commande de tâche. Pour plus d'informations, consultez la section Création d'une fenêtre de maintenance pour l'application des correctifs (p. 1141).
- Exécution manuelle `AWS-RunPatchBaseline` dans une Fonctionnalité Exécuter la commande opération . Pour plus d'informations, consultez la section Exécution des commandes depuis la console (p. 946).
- Installez les correctifs à la demande à l'aide de l'option Patch now (Corriger maintenant). Pour plus d'informations, consultez la section Application de correctifs à des instances à la demande (p. 1139).

Utilisation de Kernel Live Patching sur les instances Amazon Linux 2

Kernel Live Patching pour Amazon Linux 2 vous permet d'appliquer des correctifs de vulnérabilité de sécurité et de bogues critiques à un noyau Linux en cours d'exécution, sans redémarrer ni interrompre les applications en cours d'exécution. Cela vous permet de bénéficier d'une meilleure disponibilité des services et des applications, tout en gardant votre infrastructure sécurisée et à jour. Kernel Live Patching est pris en charge sur les instances Amazon EC2 et les machines virtuelles locales qui exécutent Amazon Linux 2.

Pour de plus amples informations sur Kernel Live Patching, veuillez consulter Kernel Live Patching sur Amazon Linux 2 dans le Amazon EC2 Guide de l'utilisateur pour les instances Linux.

Une fois que vous avez activé Kernel Live Patching sur une Amazon Linux 2 instance, vous pouvez utiliser Gestionnaire de correctifs pour appliquer des correctifs live du noyau à l'instance. L'utilisation d' Gestionnaire de correctifs est une alternative à l'utilisation de flux de travail yum existants sur l'instance pour appliquer les mises à jour.

Avant de commencer

Pour utiliser Gestionnaire de correctifs afin d'appliquer des correctifs live du noyau à vos Amazon Linux 2 instances, vérifiez que vos instances sont basées sur l'architecture et la version correctes du noyau. Pour de plus amples informations, veuillez consulter Configurations prises en charge et conditions préalables dans le Amazon EC2 Guide de l'utilisateur pour les instances Linux.

Rubriques

- À propos Kernel Live Patching d' et Gestionnaire de correctifs d' (p. 1118)
- Fonctionnement (p. 1118)

- Activation de Kernel Live Patching à l'aide de Fonctionnalité Exécuter la commande (p. 1119)
- Application des correctifs à chaud du noyau à l'aide de Fonctionnalité Exécuter la commande (p. 1121)
- Désactivation de Kernel Live Patching à l'aide de Fonctionnalité Exécuter la commande (p. 1122)

À propos Kernel Live Patching d' et Gestionnaire de correctifs d'

Mise à jour de la version du noyau

Vous n'avez pas besoin de redémarrer une instance après avoir appliqué une mise à jour du correctif live du noyau. Cependant, AWS fournit des correctifs live du noyau pour une version du noyau Amazon Linux 2 jusqu'à trois mois après sa publication. Après la période de 3 mois, vous devez effectuer une mise à jour vers une version ultérieure du noyau pour continuer à recevoir les correctifs à chaud du noyau. Nous vous recommandons d'utiliser une fenêtre de maintenance pour planifier un redémarrage de votre instance au moins une fois tous les trois mois pour demander la mise à jour de la version du noyau.

Désinstallation des correctifs live du noyau

Les correctifs Kernel Live Patching ne peuvent pas être désinstallés à l'aide d' Gestionnaire de correctifs . Au lieu de cela, vous pouvez désactiver Kernel Live Patching, ce qui supprime les packages RPM pour les correctifs live du noyau appliqués. Pour plus d'informations, consultez la section Désactivation de Kernel Live Patching à l'aide de Fonctionnalité Exécuter la commande (p. 1122).

Conformité du noyau

Dans certains cas, l'installation de tous les correctifs CVE à partir de correctifs live pour la version actuelle du noyau peut amener ce noyau dans le même état de conformité qu'une version plus récente du noyau. Lorsque cela se produit, la version la plus récente est signalée en tant que `Installed` , et l'instance est signalée en tant que `Compliant` . Cependant, aucune heure d'installation n'est signalée pour la version plus récente du noyau.

Un correctif live du noyau, plusieursCVEs

Si un correctif actif du noyau traite plusieurs CVEs , et que ceux-ci CVEs ont différentes valeurs de classification et de gravité, seules la classification et la gravité les plus élevées parmi les CVEs correctifs est signalée pour le correctif.

Le reste de cette section décrit comment utiliser Gestionnaire de correctifs pour appliquer des correctifs live du noyau aux instances qui répondent à ces exigences.

Fonctionnement

AWS publie deux types de correctifs live du noyau pour Amazon Linux 2 : les mises à jour de sécurité et les corrections de bogues. Pour appliquer ces types de correctifs, vous utilisez un document de référence de correctifs qui cible uniquement les classifications et les niveaux de gravité répertoriés dans le tableau suivant.

Classement	Gravité
`Security`	`Critical`, `Important`
`Bugfix`	`All`

Vous pouvez créer une ligne de base de correctifs personnalisée qui cible uniquement ces correctifs, ou utiliser la ligne de base de correctifs `AWS-AmazonLinux2DefaultPatchBaseline` prédéfinie. En d'autres termes, vous pouvez utiliser `AWS-AmazonLinux2DefaultPatchBaseline` avec les instances

Amazon Linux 2 sur lesquelles Kernel Live Patching est activé, et les mises à jour en direct du noyau seront appliquées.

> **Note**
>
> La `AWS-AmazonLinux2DefaultPatchBaseline` configuration spécifie une période d'attente de sept jours après la publication d'un correctif avant son installation automatique. Si vous ne souhaitez pas attendre sept jours que les correctifs live du noyau soient approuvés automatiquement, vous pouvez créer et utiliser une référence de correctifs personnalisée. Dans votre référence de correctifs, vous pouvez spécifier aucune période d'attente d'approbation automatique, ou spécifier une période plus courte ou plus longue. Pour plus d'informations, consultez la section Utilisation de références de correctifs personnalisées (p. 1127).

Nous vous recommandons la stratégie suivante pour corriger vos instances avec les mises à jour en direct du noyau :

1. Activez Kernel Live Patching sur vos instances Amazon Linux 2.
2. Utilisez Fonctionnalité Exécuter la commande pour exécuter une `Scan` opération sur vos instances à l'aide de la référence de correctifs prédéfinie `AWS-AmazonLinux2DefaultPatchBaseline` ou personnalisée qui cible également uniquement les `Security` mises à jour avec une gravité classée comme `Critical` et `Important` , et la `Bugfix` gravité `.All`
3. Ouvrez Conformité Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/compliance et vérifiez si la non-conformité pour l'application de correctifs est signalée pour l'une des instances analysées. Si c'est le cas, affichez les détails de conformité de l'instance pour déterminer si des correctifs live du noyau sont manquants dans l'instance.
4. Pour installer les correctifs live du noyau manquants, utilisez Fonctionnalité Exécuter la commande avec la même ligne de base que celle spécifiée précédemment, mais cette fois exécutez une opération `Install` au lieu d'une opération `Scan`.

   Comme les correctifs live du noyau sont installés sans avoir à redémarrer, vous pouvez choisir l'option de redémarrage `NoReboot` pour cette opération.

   > **Note**
   >
   > Vous pouvez toujours redémarrer l'instance si nécessaire pour d'autres types de correctifs installés sur l'instance, ou si vous souhaitez effectuer une mise à jour vers un noyau plus récent. Dans ces cas, choisissez plutôt l'option de redémarrage `RebootIfNeeded`.
5. Revenez à Conformité Systems Manager pour vérifier que les correctifs live du noyau ont été installés.

Activation de Kernel Live Patching à l'aide de Fonctionnalité Exécuter la commande

Pour activer Kernel Live Patching, vous pouvez exécuter des commandes `yum` sur vos instances ou utiliser Fonctionnalité Exécuter la commande et un document Systems Manager personnalisé que vous créez.

Pour de plus amples informations sur l'activation de Kernel Live Patching en exécutant des commandes `yum` directement sur l'instance, veuillez consulter Activation de Kernel Live Patching dans le Amazon EC2 Guide de l'utilisateur pour les instances Linux.

> **Note**
>
> Lorsque vous activez Kernel Live Patching, si le noyau déjà en cours d'exécution sur l'instance est antérieur à (version minimale prise en charge), le processus installe la dernière version du noyau disponible et redémarre l'instance.`kernel-4.14.165-131.185.amzn2.x86_64` Si l'instance est déjà en cours d'exécution `kernel-4.14.165-131.185.amzn2.x86_64` ou ultérieure, le processus n'installe pas une version plus récente et ne redémarre pas l'instance.

Pour activer Kernel Live Patching à l'aide de Fonctionnalité Exécuter la commande (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fonctionnalité Exécuter la commande.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Fonctionnalité Exécuter la commande.
3. Choisissez Exécuter une commande.
4. Dans la liste Command document (Document de commande), choisissez le Systems Manager document personnalisé`AWS-ConfigureKernelLivePatching`.
5. Dans la section Paramètres de commande, indiquez si vous souhaitez redémarrer les instances dans le cadre de cette opération.
6. Pour de plus amples informations sur l'utilisation des contrôles restants de cette page, veuillez consulter Exécution des commandes depuis la console (p. 946).
7. Cliquez sur Exécuter.

Pour activer Kernel Live Patching (AWS CLI)

- Exécutez la commande suivante sur votre machine locale.

  Linux

  ```
  aws ssm send-command \
      --document-name "AWS-ConfigureKernelLivePatching" \
      --parameters "EnableOrDisable=Enable" \
      --targets "Key=instanceids,Values=instance-id"
  ```

  Windows

  ```
  aws ssm send-command ^
      --document-name "AWS-ConfigureKernelLivePatching" ^
      --parameters "EnableOrDisable=Enable" ^
      --targets "Key=instanceids,Values=instance-id"
  ```

  Remplacez *`instance-id`* par l'ID de l'Amazon Linux 2instance sur laquelle vous souhaitez activer la fonction, par exemple i-02573cafcfEXAMPLE . Pour activer la fonctionnalité sur plusieurs instances, vous pouvez utiliser l'un des formats suivants.

  - `--targets "Key=instanceids,Values=instance-id1,instance-id2"`
  - `--targets "Key=tag:tag-key,Values=tag-value"`

  Pour de plus amples informations sur les autres options possibles avec la commande, veuillez consulter send-command dans le AWS CLI Command Reference.

Application des correctifs à chaud du noyau à l'aide de Fonctionnalité Exécuter la commande

Pour appliquer des correctifs live du noyau, vous pouvez soit exécuter des commandes `yum` sur vos instances, soit utiliser Fonctionnalité Exécuter la commande et le document Systems Manager `AWS-RunPatchBaseline`.

Pour de plus amples informations sur l'application des correctifs live du noyau en exécutant des commandes `yum` directement sur l'instance, veuillez consulter Application des correctifs live du noyau dans le Amazon EC2 Guide de l'utilisateur pour les instances Linux.

Pour appliquer des correctifs live du noyau à l'aide de Fonctionnalité Exécuter la commande (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fonctionnalité Exécuter la commande.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Fonctionnalité Exécuter la commande.
3. Choisissez Exécuter une commande.
4. Dans la liste Document de commande, choisissez le document Systems Manager `AWS-RunPatchBaseline`.
5. Dans la section Paramètres de commande effectuez l'une des opérations suivantes :
   - Si vous vérifiez si de nouveaux correctifs live du noyau sont disponibles, pour Opération, choisissez `Scan` . Pour Reboot Option (Option de redémarrage), si vous ne voulez pas que vos instances redémarrent après cette opération, choisissez `NoReboot` . Une fois l'opération terminée, vous pouvez vérifier les nouveaux correctifs et l'état de conformité dans Systems Manager Conformité.
   - Si vous avez déjà vérifié la conformité des correctifs et que vous êtes prêt à appliquer les correctifs à chaud du noyau disponibles, pour Opération, choisissez `Install` . Pour Reboot Option (Option de redémarrage), si vous ne voulez pas que vos instances redémarrent après cette opération, choisissez `NoReboot` .
6. Pour de plus amples informations sur l'utilisation des contrôles restants de cette page, veuillez consulter Exécution des commandes depuis la console (p. 946).
7. Cliquez sur Exécuter.

Pour appliquer des correctifs live du noyau à l'aide de Fonctionnalité Exécuter la commande (AWS CLI)

1. Pour effectuer une opération `Scan` avant de vérifier vos résultats dans Conformité Systems Manager, exécutez la commande suivante à partir de votre machine locale.

   Linux

   ```
   aws ssm send-command \
    --document-name "AWS-RunPatchBaseline" \
    --targets "Key=InstanceIds,Values=instance-id" \
    --parameters '{"Operation":["Scan"],"RebootOption":["RebootIfNeeded"]}'
   ```

   Windows

   ```
   aws ssm send-command ^
   ```

```
 --document-name "AWS-RunPatchBaseline" ^
 --targets "Key=InstanceIds,Values=instance-id" ^
 --parameters {\"Operation\":[\"Scan\"],\"RebootOption\":[\"RebootIfNeeded\"]}
```

Pour de plus amples informations sur les autres options possibles avec la commande, veuillez consulter send-command dans le AWS CLI Command Reference.

2. Pour effectuer une opération `Install` après avoir vérifié vos résultats dans Conformité Systems Manager, exécutez la commande suivante à partir de votre machine locale.

   Linux

```
aws ssm send-command \
 --document-name "AWS-RunPatchBaseline" \
 --targets "Key=InstanceIds,Values=instance-id" \
 --parameters '{"Operation":["Install"],"RebootOption":["NoReboot"]}'
```

   Windows

```
aws ssm send-command ^
 --document-name "AWS-RunPatchBaseline" ^
 --targets "Key=InstanceIds,Values=instance-id" ^
 --parameters {\"Operation\":[\"Install\"],\"RebootOption\":[\"NoReboot\"]}
```

Dans les deux commandes précédentes, remplacez *instance-id* avec l'ID de l'Amazon Linux 2instance sur laquelle vous souhaitez appliquer les correctifs live du noyau, par exemple i-02573cafcfEXAMPLE . Pour activer la fonctionnalité sur plusieurs instances, vous pouvez utiliser l'un des formats suivants.

- `--targets "Key=instanceids,Values=`*instance-id1*`,`*instance-id2*`"`
- `--targets "Key=tag:`*tag-key*`,Values=`*tag-value*`"`

Pour de plus amples informations sur les autres options que vous pouvez utiliser dans ces commandes, veuillez consulter send-command dans le AWS CLI Command Reference.

Désactivation de Kernel Live Patching à l'aide de Fonctionnalité Exécuter la commande

Pour désactiver Kernel Live Patching , vous pouvez exécuter `yum` des commandes sur vos instances ou utiliser Fonctionnalité Exécuter la commande et le Systems Manager document personnalisé `AWS-ConfigureKernelLivePatching` .

> **Note**
>
> Si vous n'avez plus besoin d'utiliser Kernel Live Patching, vous pouvez le désactiver à tout moment. Dans la plupart des cas, la désactivation de la fonctionnalité n'est pas nécessaire.

Pour de plus amples informations sur la désactivation de Kernel Live Patching en exécutant des commandes `yum` directement sur l'instance, veuillez consulter Activation de Kernel Live Patching dans le Amazon EC2 Guide de l'utilisateur pour les instances Linux.

> **Note**
>
> Lorsque vous désactivez Kernel Live Patching, le processus désinstalle le plug-in Kernel Live Patching, puis redémarre l'instance.

Pour désactiver Kernel Live Patching à l'aide de Fonctionnalité Exécuter la commande (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fonctionnalité Exécuter la commande.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Fonctionnalité Exécuter la commande.
3. Choisissez Exécuter une commande.
4. Dans la liste Document de commande, choisissez le document Systems Manager `AWS-ConfigureKernelLivePatching`.
5. Dans la section Command parameters (Paramètres de la commande), indiquez des valeurs pour les paramètres requis.
6. Pour de plus amples informations sur l'utilisation des contrôles restants de cette page, veuillez consulter Exécution des commandes depuis la console (p. 946).
7. Cliquez sur Exécuter.

Pour désactiver l'application de correctifs en direct du noyau (AWS CLI)

- Exécutez une commande similaire à la suivante.

  Linux

  ```
  aws ssm send-command \
      --document-name "AWS-ConfigureKernelLivePatching" \
      --targets "Key=instanceIds,Values=instance-id" \
      --parameters "EnableOrDisable=Disable"
  ```

  Windows

  ```
  aws ssm send-command ^
      --document-name "AWS-ConfigureKernelLivePatching" ^
      --targets "Key=instanceIds,Values=instance-id" ^
      --parameters "EnableOrDisable=Disable"
  ```

  Remplacez *`instance-id`* par l'ID de l'Amazon Linux 2instance sur laquelle vous souhaitez désactiver la fonction, par exemple i-02573cafcfEXAMPLE . Pour désactiver la fonctionnalité sur plusieurs instances, vous pouvez utiliser l'un des formats suivants.

  - `--targets "Key=instanceids,Values=instance-id1,instance-id2"`
  - `--targets "Key=tag:tag-key,Values=tag-value"`

  Pour de plus amples informations sur les autres options possibles avec la commande, veuillez consulter send-command dans le AWS CLI Command Reference.

Utilisation d' Gestionnaire de correctifs (console)

Pour utiliser Gestionnaire de correctifs , effectuez les tâches suivantes. Ces tâches sont décrites plus en détail dans cette section.

1. Vérifiez que la référence de correctifs AWS prédéfinie pour chaque type de système d'exploitation que vous utilisez répond à vos besoins. Si ce n'est pas le cas, créez une référence de correctifs qui définit un ensemble de correctifs standard pour ce type d'instance et définissez-la comme valeur par défaut.
2. Organisez les instances en groupes de correctifs en utilisant les balises Amazon EC2 (facultatif, mais recommandé).
3. Planifiez l'application des correctifs à l'aide d'une fenêtre de maintenance qui définit les instances à corriger et le moment où cette opération doit être effectuée.

   -ou-

   Instances de correctifs ou d'analyse à la demande chaque fois que cela est nécessaire.
4. Surveillez l'application des correctifs pour vérifier la conformité et enquêter sur les défaillances.

Contenu connexe

- Pour obtenir un exemple de création d'une référence de correctifs, de groupes de correctifs et d'une fenêtre de maintenant à l'aide d'AWS CLI, veuillez consulter Procédure : Appliquer un correctif à un environnement serveur (AWS CLI) (p. 1168).
- Pour de plus amples informations sur les fenêtres de maintenance, veuillez consulter AWS Systems Manager Fenêtres de maintenance (p. 688).
- Pour de plus amples informations sur la surveillance de la conformité des correctifs, veuillez consulter A propos de la conformité des correctifs (p. 783).

Rubriques

- Affichage des références de correctifs prédéfinies AWS (p. 1124)
- Affichage des résultats de conformité des correctifs (p. 1125)
- Affichage des correctifs disponibles (p. 1127)
- Utilisation de références de correctifs personnalisées (p. 1127)
- Définition d'une référence de correctifs existante comme référence par défaut (p. 1134)
- Création d'une configuration d'application de correctifs (console) (p. 1135)
- Création d'un groupe de correctifs (p. 1137)
- Application de correctifs à des instances à la demande (p. 1139)
- Création d'une fenêtre de maintenance pour l'application des correctifs (p. 1141)
- Intégration d’Gestionnaire de correctifs à AWS Security Hub (p. 1145)

Affichage des références de correctifs prédéfinies AWS

Gestionnaire de correctifsinclut une référence de correctifs prédéfinie pour chaque système d'exploitation pris en charge par Gestionnaire de correctifs . Vous pouvez soit utiliser ces références de correctifs (impossibles à personnaliser), soit en créer. La procédure suivante décrit comment afficher une référence de correctifs prédéfinie afin de voir si elle répond à vos besoins. Pour en savoir plus sur les références de correctifs, consultez À propos des références de correctifs prédéfinies et personnalisées (p. 1098).

Pour afficher les références de correctifs prédéfinies AWS

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le volet de navigation, choisissez Gestionnaire de correctifs.

   -ou-

Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Gestionnaire de correctifs.

3. Dans la liste des références de correctifs, choisissez l'ID de l'une des références de correctifs prédéfinies.

   Note

   Pour Windows Server, deux références de correctifs prédéfinies sont fournies. La référence de correctifs `AWS-WindowsPredefinedPatchBaseline-OS` prend en charge uniquement les mises à jour du système d'exploitation sur le système d'exploitation Windows lui-même. Elle est utilisée comme référence de correctifs par défaut pour les instances Windows Server, sauf si vous spécifiez une autre référence de correctifs. L'autre référence de correctifs Windows prédéfinie, `AWS-WindowsPredefinedPatchBaseline-OS-Applications`, peut être utilisée pour appliquer des correctifs au système d'exploitation Windows Server et aux applications Microsoft prises en charge.
   Pour plus d'informations, consultez la section Définition d'une référence de correctifs existante comme référence par défaut (p. 1134).

4. Choisissez l'onglet Approval rules (Règles d'approbation) et consultez la configuration de la référence de correctifs.
5. Si la configuration est acceptable pour vos instances, vous pouvez passer directement à la procédure Création d'un groupe de correctifs (p. 1137).

   -ou-

   Pour créer votre propre référence de correctifs par défaut, passez à la rubrique Utilisation de références de correctifs personnalisées (p. 1127).

Affichage des résultats de conformité des correctifs

Utilisez cette procédure pour afficher les informations de conformité des correctifs sur vos instances gérées.

Cette procédure s'applique aux opérations de correctif qui utilisent le `AWS-RunPatchBaseline` document. Pour plus d'informations sur l'affichage des informations de conformité des correctifs pour les opérations de correctif qui utilisent le `AWS-RunPatchBaselineAssociation` document, consultez Identification des instances hors conformité (p. 1110) .

Note

Les processus de configuration AWS Systems Manager rapide et AWS Systems Manager Explorer d'analyse des correctifs utilisent le `AWS-RunPatchBaselineAssociation` document.

Identifier la solution de correctif pour un problème CVE spécifique (Linux)

Pour de nombreux systèmes d'exploitation Linux, les résultats de conformité des correctifs indiquent les problèmes de bulletin de vulnérabilités et d'exposition courantes (CVE) qui sont résolus par quels correctifs. Ces informations peuvent vous aider à déterminer l'urgence à laquelle vous avez besoin d'installer un correctif manquant ou ayant échoué.

Des détails CVE sont inclus pour les types de système d'exploitation suivants :

- Amazon Linux
- Amazon Linux 2
- Oracle Linux
- Red Hat Enterprise Linux (RHEL)
- SUSE Linux Enterprise Server (SLES)

- CentOS

  Par défaut, CentOS ne fournit pas d'informations CVE sur les mises à jour. Vous pouvez toutefois activer cette prise en charge en utilisant des référentiels tiers tels que le référentiel EPEL (Extra Packages for Enterprise Linux) publié par Fedora. Pour plus d'informations, consultez EPEL sur le Wiki Fedora.

  Note

  Vous pouvez également ajouter CVE IDs à vos listes de correctifs approuvés ou rejetés dans vos références de correctifs, comme la situation et vos objectifs d'application de correctifs le justifient. Pour plus d'informations sur l'utilisation de listes de correctifs approuvés et rejetés, consultez les rubriques suivantes :

  - Utilisation de références de correctifs personnalisées (p. 1127)
  - À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés (p. 1103)
  - Fonctionnement des règles de référence de correctif sur les systèmes basés sur Linux (p. 1063)
  - Installation des correctifs (p. 1056)

Pour afficher les résultats de conformité des correctifs

1. Effectuez l'une des actions suivantes :

   Option 1 – Navigation à partir de la conformité :

   - Dans le panneau de navigation, choisissez Compliance (Conformité).

     -ou-

     Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Compliance (Conformité) dans le panneau de navigation.
   - Pour un récapitulatif des ressources de conformité, choisissez un nombre dans la colonne correspondant aux types de ressources de correctifs que vous souhaitez examiner, par exemple les ressources non conformes.
   - Dans la liste des ressources, choisissez l'ID de l'instance pour laquelle vous souhaitez examiner les résultats de conformité des correctifs.

   Option 2 – Navigation à partir d'instances gérées.

   - Dans le panneau de navigation, choisissez Instances gérées.

     -ou-

     Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Instances gérées.
   - Dans l'onglet Instances gérées, choisissez l'ID de l'instance pour laquelle vous souhaitez vérifier les résultats de conformité des correctifs.
2. Choisissez l'onglet Correctif.
3. (Facultatif) Dans la zone de recherche (🔍), choisissez parmi les filtres suivants :

   - Classement
   - Ko

- Gravité
- State

4. Choisissez l'une des valeurs disponibles pour le type de filtre que vous avez choisi. Par exemple, si vous avez choisi State (État), choisissez maintenant un état de conformité tel que Failed (Échec) ou Missing (Manquant).
5. En fonction de l'état de conformité de l'instance, vous pouvez choisir l'action à entreprendre pour remédier aux instances non conformes.

   Par exemple, vous pouvez choisir de corriger immédiatement vos instances non conformes. Pour plus d'informations sur l'application de correctifs à vos instances à la demande, consultez Application de correctifs à des instances à la demande (p. 1139) .

   Pour plus d'informations sur les états de conformité des correctifs, consultez Présentation des valeurs d'état de conformité des correctifs (p. 1113) .

Affichage des correctifs disponibles

Utilisez cette procédure pour afficher tous les correctifs disponibles pour un système d'exploitation spécifié et, éventuellement, une version spécifique du système d'exploitation.

Pour afficher les correctifs disponibles

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le volet de navigation, choisissez Gestionnaire de correctifs.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Gestionnaire de correctifs.
3. Choisissez l'onglet Correctifs.
4. Pour Operating system (Système d'exploitation), choisissez le système d'exploitation pour lequel vous souhaitez afficher les correctifs disponibles, comme ou `Windows` .Amazon Linux
5. (Facultatif) Pour Product (Produit), choisissez une version du système d'exploitation, telle que ou `WindowsServer2019` .`AmazonLinux2018.03`
6. (Facultatif) Pour ajouter ou supprimer des colonnes d'informations pour vos résultats, choisissez le bouton de configuration (⚙) en haut à droite de la liste des correctifs. (Par défaut, l'onglet Correctifs affiche des colonnes uniquement pour certaines des métadonnées de correctifs disponibles.)

   Pour plus d'informations sur les types de métadonnées que vous pouvez ajouter à votre affichage, consultez Correctif dans le Référence d'API AWS Systems Manager .

Utilisation de références de correctifs personnalisées

Gestionnaire de correctifsinclut une référence de correctifs prédéfinie pour chaque système d'exploitation pris en charge par Gestionnaire de correctifs . Vous pouvez soit utiliser ces références de correctifs (impossibles à personnaliser), soit en créer.

Les procédures suivantes décrivent comment créer, mettre à jour et supprimer vos propres références de correctifs personnalisées. Pour en savoir plus sur les références de correctifs, consultez À propos des références de correctifs prédéfinies et personnalisées (p. 1098).

Rubriques

- Création d'une référence de correctifs personnalisée (Windows) (p. 1128)

- Création d'une référence de correctifs personnalisée (Linux) (p. 1130)
- Mise à jour ou suppression d'une référence de correctifs personnalisée (console) (p. 1133)

Création d'une référence de correctifs personnalisée (Windows)

Utilisez la procédure suivante pour créer une référence de correctifs personnalisée pour les instances Windows. Pour de plus amples informations sur la création d'une référence de correctifs pour les instances Linux, veuillez consulter Création d'une référence de correctifs personnalisée (Linux) (p. 1130).

Pour obtenir un exemple de création d'une référence de correctifs limitée à l'installation des Service Packs Windows uniquement, veuillez consulter Procédure : Créer une référence de correctifs pour l'installation des Service Packs Windows (console) (p. 1166).

Pour créer une référence de correctifs personnalisée (Windows)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le volet de navigation, choisissez Gestionnaire de correctifs.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Gestionnaire de correctifs.
3. Choisissez Créer une référence de correctif.
4. Pour Name (Nom), entrez un nom pour votre nouvelle référence de correctifs, par exemple : **`MyWindowsPatchBaseline`**.
5. (Facultatif) Pour Description, saisissez une description pour cette référence de correctif.
6. Pour Operating system (Système d'exploitation), choisissez `Windows`.
7. Si vous souhaitez commencer à utiliser cette référence de correctif comme valeur par défaut pour Windows dès sa création, sélectionnez Définir cette référence de correctif comme référence par défaut pour les instances Windows Server.

   Si vous choisissez de ne pas définir cette référence de correctifs pour une utilisation immédiate, vous pouvez le faire plus tard. Pour plus d'informations, consultez la section Définition d'une référence de correctifs existante comme référence par défaut (p. 1134).
8. Dans la section Règles d'approbation pour les systèmes d'exploitation), utilisez les champs pour créer une ou plusieurs règles d'approbation automatique.

   - Produit : Version des systèmes d'exploitation à laquelle la règle d'approbation s'applique, par exemple `WindowsServer2008` . La sélection par défaut est `All` .
   - Classification : Le type de correctifs auquel la règle d'approbation s'applique, par exemple `CriticalUpdates` , et `Drivers` .`Tools` La sélection par défaut est `All` .

     > **Tip**
     >
     > Vous pouvez inclure des installations de Service Packs Windows dans vos règles d'approbation en incluant `ServicePacks` ou en choisissant `All` dans votre liste Classification. Pour obtenir un exemple, veuillez consulter Procédure : Créer une référence de correctifs pour l'installation des Service Packs Windows (console) (p. 1166).
   - Gravité : Valeur de gravité des correctifs à laquelle la règle va s'appliquer, par exemple `Critical` . La sélection par défaut est `All` .
   - Approbation automatique : Méthode de sélection des correctifs pour approbation automatique.
     - Approuvez les correctifs après un nombre de jours spécifié : Nombre de jours qu' doit attendre après la publication dGestionnaire de correctifs'un correctif avant qu'il soit automatiquement approuvé. Vous pouvez entrer tout nombre entier situé entre zéro (0) et 100.

- Approuvez les correctifs publiés jusqu'à une date spécifique : Date de publication des correctifs pour laquelle applique Gestionnaire de correctifs automatiquement tous les correctifs publiés à cette date ou avant cette date. Par exemple, si vous spécifiez le 7 juillet 2020, aucun correctif publié à partir du 8 juillet 2020 n'est installé automatiquement.
- (Facultatif) Rapport de conformité : Niveau de gravité que vous voulez qu' attribue aux correctifs approuvés par la référence, comme `High` .

  Note

  Si un correctif approuvé est signalé comme manquant, l'option que vous choisissez dans Compliance reporting (Rapport de conformité), comme `Critical` ou `Medium`, détermine la gravité de la violation de conformité.

9. Dans la section Règles d'approbation pour les applications Microsoft, utilisez les champs pour créer une ou plusieurs règles d'approbation automatique.
   - Famille de produits : La famille de produits Microsoft générale pour laquelle vous souhaitez spécifier une règle, telle que Office ou Exchange Server.
   - Produit : Version de l'application à laquelle la règle d'approbation s'applique, telle que Office 2016 ou Active Directory Rights Management Services Client 2.0 2016. La sélection par défaut est All.
   - Classification : Type de correctifs auquel s'applique la règle d'approbation, par exemple `CriticalUpdates` . La sélection par défaut est `All` .
   - Gravité : Valeur de gravité des correctifs à laquelle la règle s'applique, par exemple `Critical` . La sélection par défaut est `All` .
   - Approbation automatique : Méthode de sélection des correctifs pour approbation automatique.
     - Approuvez les correctifs après un nombre de jours spécifié : Nombre de jours qu' doit attendre après la publication dGestionnaire de correctifs'un correctif avant qu'un correctif soit automatiquement approuvé. Vous pouvez entrer tout nombre entier situé entre zéro (0) et 100.
     - Approuvez les correctifs publiés jusqu'à une date spécifique : Date de publication des correctifs pour laquelle applique Gestionnaire de correctifs automatiquement tous les correctifs publiés à cette date ou avant cette date. Par exemple, si vous spécifiez le 7 juillet 2020, aucun correctif publié à partir du 8 juillet 2020 n'est installé automatiquement.
   - (Facultatif) Rapport de conformité : Niveau de gravité que vous voulez affecter aux correctifs approuvés par la référence, par exemple `High` .

     Note

     Si un correctif approuvé est signalé comme manquant, l'option que vous choisissez dans Compliance reporting (Rapport de conformité), comme `Critical` ou `Medium`, détermine la gravité de la violation de conformité.
10. Si vous souhaitez explicitement approuver des correctifs en plus de ceux conformes à vos règles d'approbation, procédez comme suit dans la section Patch exceptions (Exceptions de correctifs) :
    - Pour Approved patches (Correctifs approuvés), entrez une liste séparée par des virgules des correctifs à approuver.

      Note

      Pour plus d'informations sur les formats acceptés pour les listes de correctifs approuvés et de correctifs rejetés, consultez À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés (p. 1103).
    - (Facultatif) Pour Approved patches compliance level (Niveau de conformité des correctifs approuvés), affectez un niveau de conformité aux correctifs figurant dans la liste.
11. Si vous souhaitez rejeter explicitement des correctifs conformes par ailleurs à vos règles d'approbation, procédez comme suit dans la section Patch exceptions (Exceptions de correctifs) :

- Pour Rejected patches (Correctifs rejetés), entrez une liste séparée par des virgules des correctifs à rejeter.

  Note

  Pour plus d'informations sur les formats acceptés pour les listes de correctifs approuvés et de correctifs rejetés, consultez À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés (p. 1103).
- Pour Rejected patches action (Action pour les correctifs rejetés), sélectionnez l'action qu' doit effectuer sur les correctifs inclus dans la Gestionnaire de correctifs liste des correctifs rejetés.
  - Autorisez en tant que dépendance : Un package de la liste Correctifs rejetés est installé uniquement s'il constitue une dépendance d'un autre package. Il est considéré conforme à la référence de correctifs et son statut est signalé comme InstalledOther . Il s'agit de l'action par défaut si aucune option n'est spécifiée.
  - Bloc : Les packages de la liste Correctifs rejetés et les packages qui les incluent en tant que dépendances ne sont pas installés, quelles que soient les circonstances. Si un package a été installé avant d'être ajouté à la liste Correctifs rejetés, il est considéré comme non conforme à la référence de correctifs et son statut est signalé comme InstalledRejected .

12. (Facultatif) Pour Gérer les balises, appliquez une ou plusieurs paires nom/valeur de clé de balise à la référence de correctifs.

    Les balises sont des métadonnées facultatives que vous affectez à une ressource. Les balises vous permettent de classer une ressource de différentes façons, par exemple, par objectif, par propriétaire ou par environnement. Par exemple, vous pouvez baliser une référence de correctifs pour identifier le niveau de gravité de correctifs qu'elle spécifie, la famille de systèmes d'exploitation à laquelle elle s'applique et le type d'environnement. Dans ce cas, vous pouvez spécifier des balises similaires aux paires nom/valeur de clé suivantes :

    - `Key=PatchSeverity,Value=Critical`
    - `Key=OS,Value=RHEL`
    - `Key=Environment,Value=Production`
13. Choisissez Créer une référence de correctif.

Création d'une référence de correctifs personnalisée (Linux)

Utilisez la procédure suivante pour créer une référence de correctifs personnalisée pour les instances Linux. Pour de plus amples informations sur la création d'une référence de correctifs pour les instances Windows Server, veuillez consulter Création d'une référence de correctifs personnalisée (Windows) (p. 1128).

Pour créer une référence de correctifs personnalisée pour les instances Linux

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le volet de navigation, choisissez Gestionnaire de correctifs.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Gestionnaire de correctifs.
3. Choisissez Créer une référence de correctif.
4. Pour Name (Nom), entrez un nom pour votre nouvelle référence de correctifs, par exemple : **`MyRHELPatchBaseline`**.
5. (Facultatif) Pour Description, saisissez une description pour cette référence de correctif.

6. Pour Operating system (Système d'exploitation), choisissez un système d'exploitation, par exemple, `Red Hat Enterprise Linux`.
7. Si vous souhaitez commencer à utiliser cette référence de correctifs comme référence par défaut pour le système d'exploitation sélectionné dès sa création, cochez la case en regard de Définir cette référence de correctifs comme référence par défaut pour ***`operating system name`*** Instances .

   Pour de plus amples informations, sur la définition d'une référence de correctifs existante en tant que référence par défaut, veuillez consulter Définition d'une référence de correctifs existante comme référence par défaut (p. 1134).
8. Dans la section Règles d'approbation pour les systèmes d'exploitation), utilisez les champs pour créer une ou plusieurs règles d'approbation automatique.

   - Produit : Version des systèmes d'exploitation à laquelle s'applique la règle d'approbation, par exemple `RedhatEnterpriseLinux7.4` . La sélection par défaut est `All` .
   - Classification : Type de correctifs auquel s'applique la règle d'approbation, par exemple `Security` ou `Enhancement` . La sélection par défaut est `All` .

     Tip

     Vous pouvez configurer une référence de correctifs pour contrôler si des mises à niveau mineures pour Linux sont installées, par exemple RHEL 7.8. Des mises à niveau de version mineure peuvent être installées automatiquement à Gestionnaire de correctifs condition que la mise à jour soit disponible dans le référentiel approprié.
     Pour les systèmes d'exploitation Linux, les mises à niveau de versions mineures ne sont pas classées de manière cohérente. Elles peuvent être classées comme correctifs de bogues ou mises à jour de sécurité, ou non classés, même dans la même version du noyau. Voici quelques options pour vérifier si une référence de correctifs les installe.

     - Option 1 : La règle d'approbation la plus large pour s'assurer que des mises à niveau de version mineure sont installées lorsqu'elles sont disponibles consiste à spécifier Classification en tant que **`All`** (*) et à choisir l'option Include nonsecurity updates (Inclure les mises à jour non liées à la sécurité).
     - Option 2 : Pour vous assurer que les correctifs pour une version de système d'exploitation sont installés, vous pouvez utiliser un caractère générique (*) pour spécifier son format de noyau dans la section des exceptions de correctif de la référence. Par exemple, le format du noyau pour RHEL 7.* est `kernel-3.10.0-*.el7.x86_64`.

       Entrez `kernel-3.10.0-*.el7.x86_64` dans la liste Correctifs approuvés de votre référence de correctifs pour vous assurer que tous les correctifs, y compris les mises à niveau mineures de version, sont appliqués à vos instances RHEL 7.* (Si vous connaissez le nom exact du package d'un correctif de version mineur, saisissez-le à la place de cette valeur.)
     - Option 3 : Vous pouvez avoir le plus de contrôle sur les correctifs appliqués à vos instances gérées, y compris les mises à niveau de version mineure, en utilisant le InstallOverrideList (p. 1083) paramètre du `AWS-RunPatchBaseline` document. Pour plus d'informations, consultez la section À propos du Document SSM AWS-RunPatchBaseline (p. 1080).
   - Gravité : Valeur de gravité des correctifs à laquelle la règle va s'appliquer, par exemple `Critical` . La sélection par défaut est `All` .
   - Approbation automatique : Méthode de sélection des correctifs pour approbation automatique.

     Note

     Comme il n'est pas possible de déterminer de manière fiable les dates de publication des packages de mise à jour pour Ubuntu Server, les options d'approbation automatique ne sont pas prises en charge pour ce système d'exploitation.

- Approuvez les correctifs après un nombre de jours spécifié : Nombre de jours qu' doit attendre après la publication dGestionnaire de correctifs'un correctif avant qu'un correctif soit automatiquement approuvé. Vous pouvez entrer tout nombre entier situé entre zéro (0) et 100.
- Approuvez les correctifs publiés jusqu'à une date spécifique : Date de publication des correctifs pour laquelle applique Gestionnaire de correctifs automatiquement tous les correctifs publiés à cette date ou avant cette date. Par exemple, si vous spécifiez le 7 juillet 2020, aucun correctif publié à partir du 8 juillet 2020 n'est installé automatiquement.

- (Facultatif) Rapport de conformité : Niveau de gravité que vous voulez affecter aux correctifs approuvés par la référence, par exemple `High` .

  Note

  Si un correctif approuvé est signalé comme manquant, l'option que vous choisissez dans Compliance reporting (Rapport de conformité), comme `Critical` ou `Medium`, détermine la gravité de la violation de conformité.

- Inclure les mises à jour non liées à la sécurité : Cochez la case pour installer les correctifs non liés à la sécurité du système d'exploitation Linux disponibles dans le référentiel source, en plus des correctifs liés à la sécurité.

  Note

  Pour SUSE Linux Enterprise Server (SLES), il n'est pas nécessaire de cocher cette case, car les correctifs pour les problèmes liés ou non à la sécurité sont installés par défaut sur les instances SLES. Pour plus d'informations, consultez le contenu relatif à SLES dans Sélection des correctifs de sécurité (p. 1049).

Pour en savoir plus sur l'utilisation des règles d'approbation dans une référence de correctifs personnalisée, consultez À propos des références personnalisées (p. 1101).

9. Si vous souhaitez explicitement approuver des correctifs en plus de ceux conformes à vos règles d'approbation, procédez comme suit dans la section Patch exceptions (Exceptions de correctifs) :

   - Pour Approved patches (Correctifs approuvés), entrez une liste séparée par des virgules des correctifs à approuver.

     Note

     Pour plus d'informations sur les formats acceptés pour les listes de correctifs approuvés et de correctifs rejetés, consultez À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés (p. 1103).

   - (Facultatif) Pour Approved patches compliance level (Niveau de conformité des correctifs approuvés), affectez un niveau de conformité aux correctifs figurant dans la liste.
   - Si des correctifs approuvés que vous spécifiez ne sont pas liés à la sécurité, cochez la case Approved patches include non-security updates (Les correctifs approuvés incluent des mises à jour non liées à la sécurité) pour que ces correctifs soient aussi installés sur votre système d'exploitation Linux.

10. Si vous souhaitez rejeter explicitement des correctifs conformes par ailleurs à vos règles d'approbation, procédez comme suit dans la section Patch exceptions (Exceptions de correctifs) :

    - Pour Rejected patches (Correctifs rejetés), entrez une liste séparée par des virgules des correctifs à rejeter.

      Note

      Pour plus d'informations sur les formats acceptés pour les listes de correctifs approuvés et de correctifs rejetés, consultez À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés (p. 1103).

- Pour Rejected patches action (Action pour les correctifs rejetés), sélectionnez l'action qu' doit effectuer sur les correctifs inclus dans la Gestionnaire de correctifs liste des correctifs rejetés.
  - Autorisez en tant que dépendance : Un package de la liste Correctifs rejetés est installé uniquement s'il constitue une dépendance d'un autre package. Il est considéré conforme à la référence de correctifs et son statut est signalé comme InstalledOther . Il s'agit de l'action par défaut si aucune option n'est spécifiée.
  - Bloc : Les packages de la liste Correctifs rejetés et les packages qui les incluent en tant que dépendances ne sont pas installés, quelles que soient les circonstances. Si un package a été installé avant d'être ajouté à la liste Correctifs rejetés, il est considéré comme non conforme à la référence de correctifs et son statut est signalé comme InstalledRejected .

11. (Facultatif) Si vous souhaitez spécifier d'autres référentiels de correctifs pour différentes versions d'un système d'exploitation, tels que AmazonLinux2016 .03 et AmazonLinux2017 .09, effectuez les opérations suivantes pour chaque produit dans la section Sources de correctifs :

    - Dans Name (Nom), entrez un nom qui vous aidera à identifier la configuration de la source.
    - Dans Product (Produit), sélectionnez la version des systèmes d'exploitation à laquelle est destiné le référentiel source de correctifs, comme `RedhatEnterpriseLinux7.4`.
    - Dans Configuration, entrez la valeur de la configuration du référentiel yum à utiliser. Exemple :

      ```
      [main]
      cachedir=/var/cache/yum/$basesearch$releasever
      keepcache=0
      debuglevel=2
      ```

      Choisissez Add another source (Ajouter une autre source) pour spécifier un référentiel source pour chaque version supplémentaire du système d'exploitation, jusqu'à un maximum de 20.

      Pour en savoir plus sur les autres référentiels source de correctifs, consultez Spécification d'un autre référentiel source de correctifs (Linux) (p. 1054).

12. (Facultatif) Pour Gérer les balises, appliquez une ou plusieurs paires nom/valeur de clé de balise à la référence de correctifs.

    Les balises sont des métadonnées facultatives que vous affectez à une ressource. Les balises vous permettent de classer une ressource de différentes façons, par exemple, par objectif, par propriétaire ou par environnement. Par exemple, vous pouvez baliser une référence de correctifs pour identifier le niveau de gravité de correctifs qu'elle spécifie, la famille de systèmes d'exploitation à laquelle elle s'applique et le type d'environnement. Dans ce cas, vous pouvez spécifier des balises similaires aux paires nom/valeur de clé suivantes :

    - `Key=PatchSeverity,Value=Critical`
    - `Key=OS,Value=RHEL`
    - `Key=Environment,Value=Production`

13. Choisissez Créer une référence de correctif.

Mise à jour ou suppression d'une référence de correctifs personnalisée (console)

Vous pouvez mettre à jour ou supprimer une référence de correctifs personnalisée que vous avez créée. Lorsque vous mettez à jour une référence de correctifs, vous pouvez modifier son nom, sa description, ses règles d'approbation et ses exceptions pour les correctifs approuvés et rejetés. Vous pouvez également mettre à jour les balises qui sont appliquées à la référence de correctifs. Vous ne pouvez pas modifier le type de système d'exploitation pour lequel une référence de correctifs a été créée et vous ne pouvez pas apporter des modifications à une référence de correctifs prédéfinie fournie par AWS.

Mise à jour ou suppression d'une référence de correctifs (console)

Suivez les étapes ci-dessous pour mettre à jour ou supprimer une référence de correctifs.

Pour mettre à jour ou supprimer une référence de correctifs (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le volet de navigation, choisissez Gestionnaire de correctifs.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Gestionnaire de correctifs.
3. Choisissez la référence de correctifs à mettre à jour ou supprimer, puis exécutez l'une des tâches suivantes :

   - Pour supprimer la référence de correctifs à partir de votre compte, choisissez Supprimer. Le système vous invite à confirmer vos actions.
   - Pour modifier le nom ou la description d'une référence de correctifs, les règles d'approbation ou les exceptions de correctifs, choisissez Modifier. Sur la page Modifier la référence de correctif, modifiez les valeurs et options souhaitées, puis choisissez Enregistrer les modifications.
   - Pour ajouter, modifier ou supprimer des balises appliquées à la référence de correctifs, choisissez l'onglet Balises, puis choisissez Modifier les balises. Sur la page Edit patch baseline tags (Modifier les balises de référence de correctif), mettez à jour les balises de la référence de correctifs, puis choisissez Enregistrer les modifications.

   Pour de plus amples informations sur les choix de configuration que vous pouvez effectuer, veuillez consulter Utilisation de références de correctifs personnalisées (p. 1127).

Définition d'une référence de correctifs existante comme référence par défaut

Lorsque vous créez une référence de correctifs personnalisée, vous pouvez la définir comme référence par défaut pour le type de système d'exploitation associé dès sa création. Pour plus d'informations, consultez la section Utilisation de références de correctifs personnalisées (p. 1127).

Vous pouvez également définir une référence de correctifs existante en tant que référence par défaut pour un type de système d'exploitation.

Pour définir une référence de correctifs comme référence par défaut

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le volet de navigation, choisissez Gestionnaire de correctifs.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Gestionnaire de correctifs.
3. Dans la liste des références de correctifs, choisissez le bouton correspondant à une référence de correctifs qui n'est pas actuellement définie comme référence par défaut pour un type de système d'exploitation.

Tip

La colonne Référence par défaut indique les références qui sont actuellement définies comme références par défaut.

4. Dans le menu Actions, choisissez Définir une référence de correctif par défaut.
5. Dans la boîte de dialogue de confirmation, sélectionnez Set default (Définir par défaut).

Création d'une configuration d'application de correctifs (console)

Une configuration d'application de correctifs définit une opération unique d'application de correctifs. La configuration spécifie les instances d'application de correctifs sur lesquelles la référence de correctifs doit être appliquée, le calendrier d'application des correctifs et la fenêtre de maintenance à laquelle la configuration doit être associée.

Note

De nombreux cas d'utilisation des correctifs tirent profit de l'application des correctifs à des instances selon un calendrier avec une fenêtre de maintenance, mais vous pouvez également exécuter une opération ponctuelle d'application des correctifs manuellement sans fenêtre de maintenance. Pour plus d'informations, consultez la section Application de correctifs à des instances à la demande (p. 1139).

Pour minimiser l'impact sur la disponibilité de votre serveur, nous vous recommandons de configurer une fenêtre de maintenance pour exécuter l'application des correctifs au cours de périodes qui ne perturberont pas vos opérations professionnelles. Pour de plus amples informations sur les fenêtres de maintenance, veuillez consulter AWS Systems Manager Fenêtres de maintenance (p. 688).

Si vous prévoyez d'ajouter la configuration d'application des correctifs à une fenêtre de maintenance, vous devez configurer des rôles et des autorisations pour Fenêtres de maintenance avant de commencer cette procédure. Pour plus d'informations, consultez la section Configuration d'Fenêtres de maintenance (p. 690).

Pour créer une configuration d'application de correctifs (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le volet de navigation, choisissez Gestionnaire de correctifs.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Gestionnaire de correctifs.
3. Choisissez Configure patching (Configurer l'application des correctifs).
4. Dans la section Instances to patch (Instances à corriger), choisissez l'une des actions suivantes :

   - Entrez les balises d'instance : Entrez une clé de balise et une valeur de balise facultative pour spécifier l'instance balisée à corriger. Cliquez sur Ajouter pour inclure d'autres instances balisées.
   - Sélectionnez un groupe de correctifs : Choisissez le nom d'un groupe de correctifs existant qui inclut les instances que vous souhaitez corriger.

     Note

     La liste Select a patch group (Sélectionner un groupe de correctifs) affiche uniquement les groupes de correctifs qui sont attachés ou enregistrés auprès d'une référence de correctifs. Vous pouvez enregistrer un groupe de correctifs auprès d'une référence de correctifs de deux manières différentes. Vous pouvez utiliser la commande d'interface de ligne de commande register-patch-baseline-for-patch-group ou vous pouvez afficher une référence

de correctifs dans la console Systems Manager et sélectionner Modifier les groupes de correctifs dans le menu Actions.
Sinon, pour spécifier un groupe de correctifs existant qui n'est pas enregistré auprès de la référence de correctifs, choisissez Enter instance tag, type (Entrer une balise d'instance, type) `Patch Group` comme clé de balise et le nom du groupe de correctifs comme valeur de balise.

- Sélectionnez les instances manuellement : Cochez la case en regard du nom de chaque instance à corriger.

  Note

  Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.

5. Dans la section Patching schedule (Planification d'application de correctifs), choisissez l'une des actions suivantes :
   - Sélectionnez une fenêtre de maintenance existante : Dans la liste, sélectionnez une fenêtre de maintenance que vous avez déjà créée, puis passez à l'étape 7.
   - Programme d'une nouvelle fenêtre de maintenance : Créez une nouvelle fenêtre de maintenance à associer à cette configuration d'application de correctifs.
   - Ignorer la planification et corriger maintenant : Exécutez une opération ponctuelle d'application manuelle de correctifs sans planification ni fenêtre de maintenance. Passez à l'étape 7.
6. Si vous avez choisi Schedule in a new maintenance window (Planifier dans une nouvelle fenêtre de maintenance) à l'étape 5, procédez comme suit sous How do you want to specify a patching schedule? (Comment souhaitez-vous spécifier une planification d'application de correctifs ?) :
   - Sous How do you want to specify a maintenance window schedule? (Comment souhaitez-vous spécifier une planification de fenêtre de maintenance ?), choisissez l'option de générateur de planification ou d'expression.
   - Sous Maintenance window run frequency (Fréquence d'exécution de la fenêtre de maintenance), spécifiez la fréquence à laquelle la fenêtre de maintenance s'exécute. Si vous spécifiez une expression CRON/Rate, veuillez consulter Référence : Expressions cron et rate pour Systems Manager (p. 1392) pour de plus amples informations.
   - Pour Maintenance window duration (Durée de la fenêtre de maintenance), spécifiez le nombre d'heures pendant lesquelles la fenêtre de maintenance est autorisée à s'exécuter avant d'expirer.
   - Pour Maintenance window name (Nom de la fenêtre de maintenance), entrez un nom permettant d'identifier la fenêtre de maintenance.
7. Dans la zone Patching operation (Opération d'application de correctifs), choisissez d'analyser les instances pour rechercher les correctifs manquants et les appliquer selon les besoins, ou d'analyser uniquement et de générer la liste des correctifs manquants.
8. (Facultatif) Dans la zone Additional settings (Paramètres supplémentaires), si des instances cibles que vous avez sélectionnées appartiennent à un groupe de correctifs, vous pouvez modifier la référence de correctifs qui est associée au groupe de correctifs. Pour ce faire, procédez comme suit :
   1. Choisissez le bouton en regard du nom de la référence de correctifs associée.
   2. Choisissez Change patch baseline registration (Modifier l'enregistrement de référence de correctifs).
   3. Choisissez les références de correctifs que vous souhaitez spécifier pour cette configuration en décochant et en cochant les cases en regard des noms des références de correctifs.
   4. Choisissez Fermer.

Note

Pour toutes les instances cibles que vous avez sélectionnées qui ne font pas partie d'un groupe de correctifs, utilise Gestionnaire de correctifs plutôt la référence de correctifs par défaut pour le type de système d'exploitation de l'instance.

9. Choisissez Configure patching (Configurer l'application des correctifs).

Si vous avez créé une nouvelle fenêtre de maintenance pour cette configuration d'application de correctifs, vous pouvez l'ajouter ou y apporter des modifications dans la zone Fenêtres de maintenance de Systems Manager. Pour plus d'informations, consultez la section Mettre à jour ou supprimer une fenêtre de maintenance (console) (p. 713).

Création d'un groupe de correctifs

Pour faciliter l'organisation de votre application de correctifs, nous vous recommandons d'ajouter des instances à des groupes de correctifs en utilisant les balises. Les groupes de correctifs nécessitent l'utilisation de la clé de balise Groupe de correctifs. Vous pouvez spécifier une valeur mais la clé de balise doit être Patch Group (Groupe de correctifs). Pour de plus amples informations sur les groupes de correctifs, veuillez consulter À propos des groupes de correctifs (p. 1105).

Une fois que vous avez regroupé vos instances à l'aide de balises, vous devez ajouter la valeur du groupe de correctifs à une référence de correctifs. En enregistrant le groupe de correctifs auprès d'une référence de correctifs, vous veillez à ce que les correctifs appropriés soient installés lors de l'opération d'application des correctifs.

Rubriques

- Tâche 1 : Ajouter des instances EC2 à un groupe de correctifs à l'aide de balises (p. 1137)
- Tâche 2 : Ajouter des instances gérées à un groupe de correctifs à l'aide de balises (p. 1138)
- Tâche 3 : Ajout d'un groupe de correctifs à une référence de correctifs (p. 1138)

Tâche 1 : Ajouter des instances EC2 à un groupe de correctifs à l'aide de balises

Pour les instances EC2, vous pouvez ajouter des balises à l'aide de la console AWS Systems Manager, de la console Amazon EC2, de la commande de l'AWS CLI `create-tags` ou de l'action d'API `CreateTags`.

Pour ajouter des instances EC2 à un groupe de correctifs (console AWS Systems Manager)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Instances gérées.
3. Dans la liste Instances gérées, choisissez l'ID d'une instance EC2 gérée que vous souhaitez configurer pour l'application de correctifs.

   Note

   Lors de l'utilisation de la console Amazon EC2 et de AWS CLI, il est possible d'appliquer des balises `Key = Patch Group` à des instances qui ne sont pas encore configurées pour une utilisation avec Systems Manager.
   Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.
4. Sélectionnez l'onglet Balises, puis choisissez Modifier.
5. Dans la colonne de gauche, tapez **`Patch Group`**.

6. Dans la colonne de droite, entrez une valeur qui vous aidera à comprendre quelles instances seront corrigées.
7. Choisissez Save.
8. Répétez cette procédure pour ajouter d'autres instances gérées au même groupe de correctifs.

Pour ajouter des instances EC2 à un groupe de correctifs (console Amazon EC2)

1. Ouvrez la consoleAmazon EC2, puis choisissez Instances dans le panneau de navigation.
2. Dans la liste d'instances, choisissez une instance que vous souhaitez configurer pour l'application de correctifs.
3. Dans le menu Actions, choisissez Paramètres de l'instance, Ajouter/Modifier des balises.
4. Si une ou plusieurs balises sont déjà appliquées à l'instance, choisissez Créer une balise.
5. Pour Clé, tapez **`Patch Group`**.
6. Pour Valeur, entrez une valeur qui vous aidera à comprendre quelles instances seront corrigées.
7. Choisissez Save.
8. Répétez cette procédure pour ajouter d'autres instances au même groupe de correctifs.

Tâche 2 : Ajouter des instances gérées à un groupe de correctifs à l'aide de balises

Pour les instances gérées hybrides (mi-*), vous pouvez ajouter des balises à l'aide de la AWS Systems Manager console AWS CLI , de la `add-tags-to-resource` commande de l' ou de l'action d'API `AddTagsToResource` . Vous ne pouvez pas ajouter de balises pour les instances gérées hybrides à l'aide de la Amazon EC2 console.

Pour ajouter des instances gérées à un groupe de correctifs (console AWS Systems Manager)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Instances gérées.
3. Dans la liste Instances gérées, choisissez une instance gérée que vous souhaitez configurer pour l'application de correctifs.

   Note

   Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.
4. Choisissez Afficher les détails.
5. Sélectionnez l'onglet Balises, puis choisissez Modifier.
6. Dans la colonne de gauche, tapez **`Patch Group`**.
7. Dans la colonne de droite, entrez une valeur qui vous aidera à comprendre quelles instances seront corrigées.
8. Choisissez Save.
9. Répétez cette procédure pour ajouter d'autres instances gérées au même groupe de correctifs.

Tâche 3 : Ajout d'un groupe de correctifs à une référence de correctifs

Pour associer une référence de correctifs spécifique à vos instances, vous devez ajouter la valeur du groupe de correctifs à la référence de correctifs. En enregistrant le groupe de correctifs auprès d'une référence de correctifs, vous veillez à ce que les correctifs appropriés soient installés lors de l'exécution

de l'application des correctifs. Pour de plus amples informations sur les groupes de correctifs, veuillez consulter À propos des groupes de correctifs (p. 1105).

Pour ajouter un groupe de correctifs à une référence de correctifs (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Gestionnaire de correctifs.
3. Dans la liste Références de correctifs, choisissez la référence de correctifs que vous souhaitez configurer pour votre groupe de correctifs.
4. Choisissez Actions, puis Modifier les groupes de correctifs.
5. Entrez la valeur de balise que vous avez ajoutée à vos instances gérées dans la section précédente, puis choisissez Ajouter.

Application de correctifs à des instances à la demande

A l'aide de l'option Patch now (Corriger maintenant) dans Gestionnaire de correctifs , vous pouvez exécuter des opérations d'application de correctifs à la demande à partir de la Systems Manager console . Cela signifie que vous n'avez pas besoin de créer une planification pour mettre à jour l'état de conformité de vos instances ou pour installer des correctifs sur les instances non conformes. Vous n'avez pas non plus besoin de basculer entre les Gestionnaire de correctifs zones Fenêtres de maintenance et de la Systems Manager console pour configurer ou modifier une fenêtre d'application de correctifs planifiée.

Fonctionnement de « Patch now »

Pour exécuter Patch maintenant, vous spécifiez seulement deux options :

- Indique s'il faut rechercher les correctifs manquants uniquement, ou analyser et installer les correctifs sur vos instances
- Les instances sur lesquelles exécuter l'opération

Lors de l'exécution de l'opération de correctif, elle utilise la référence de correctifs actuellement définie comme référence par défaut pour le type de système d'exploitation de vos instances. (Il peut s'agir d'une AWS référence prédéfinie ou de la référence personnalisée que vous avez définie par défaut.) En outre, les options de simultanéité et de seuil d'erreurs sont gérées par Gestionnaire de correctifs . Vous n'avez pas besoin de spécifier le nombre d'instances à corriger simultanément, ni le nombre d'erreurs autorisées avant que l'opération échoue. Gestionnaire de correctifs applique les paramètres de simultanéité et de seuil d'erreurs décrits dans les tableaux suivants lorsque vous effectuez des correctifs à la demande.

Concurrency

Nombre total d'instances dans l'opération Patch now	Nombre d'instances analysées ou corrigées à la fois
Moins de 25	1
25 à 100	5 %
101 à 1 000	8 %
Plus de 1 000	10 %

Seuil d'erreurs

Nombre total d'instances dans l'opération Patch now	Nombre d'erreurs autorisées avant l'échec de l'opération
Moins de 25	1
25 à 100	5
101 à 1 000	10
Plus de 1 000	10

Exécution de « Patch now »

Utilisez la procédure suivante pour corriger vos instances à la demande.

Pour exécuter « Patch now »

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Gestionnaire de correctifs.
3. Sur la AWS Systems Manager Gestionnaire de correctifs page ou sur la page Références de correctifs, en fonction de celle qui s'ouvre, choisissez Correctif maintenant.
4. Pour l'opération d'application de correctifs, choisissez l'une des options suivantes :
   - Analyse : Gestionnaire de correctifs détecte quels correctifs sont manquants dans vos instances, mais ne les installe pas. Vous pouvez afficher les résultats dans le tableau de bord de conformité ou dans d'autres outils que vous utilisez pour afficher la conformité des correctifs.
   - Analysez et installez : Gestionnaire de correctifs identifie les correctifs manquants au niveau de vos instances et les installe.
5. Uniquement si vous avez choisi Scan and install in the previous step (Analyser et installer à l'étape précédente) : Pour l'option Reboot (Redémarrer), spécifiez si Gestionnaire de correctifs peut redémarrer vos instances maintenant si elles le nécessitent.
6. Pour Instances à corriger, choisissez l'une des options suivantes :
   - Permet de corriger toutes les instances : Gestionnaire de correctifs exécute l'opération spécifiée sur toutes les instances gérées de votre compte dans la AWS région actuelle.
   - Pour corriger uniquement les instances cibles que je spécifie : Vous spécifiez les instances à cibler à l'étape suivante.
7. Uniquement si vous avez choisi Patch uniquement les instances cibles que je spécifie à l'étape précédente : Dans la section Target selection (Sélection de la cible), identifiez les instances sur lesquelles vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant des instances manuellement ou en spécifiant un groupe de ressources.

   Note

   Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.
   Si vous choisissez de cibler un groupe de ressources, notez que les groupes de ressources basés sur une AWS CloudFormation pile doivent toujours être balisés avec la `aws:cloudformation:stack-id` balise par défaut. Si elle a été supprimée, Gestionnaire de correctifs peut ne pas être en mesure de déterminer quelles instances appartiennent au groupe de ressources.

8. Choisissez Patch now (Correctif maintenant).

   La page récapitulative de l'exécution de l'association s'ouvre. (Le correctif utilise désormais State Manager des associations pour ses opérations.) Dans la zone Operation summary (Récapitulatif de l'opération), vous pouvez surveiller le statut de l'analyse ou de l'application de correctifs sur les instances que vous avez spécifiées.

Création d'une fenêtre de maintenance pour l'application des correctifs

> **Important**
>
> Vous pouvez continuer à utiliser cette rubrique existante pour créer une fenêtre de maintenance afin d'appliquer des correctifs. Toutefois, nous vous recommandons d'utiliser la page Configure patching (Configuration de l'application des correctifs) à la place. Pour plus d'informations, consultez la section Création d'une configuration d'application de correctifs (console) (p. 1135). Et bien que de nombreux cas d'utilisation des correctifs bénéficient de l'application des correctifs à des instances selon un calendrier avec une fenêtre de maintenance, vous pouvez également exécuter une opération ponctuelle d'application des correctifs manuellement sans fenêtre de maintenance. Pour plus d'informations, consultez la section Application de correctifs à des instances à la demande (p. 1139).

Pour minimiser l'impact sur la disponibilité de votre serveur, nous vous recommandons de configurer une fenêtre de maintenance pour exécuter l'application des correctifs au cours de périodes qui ne perturberont pas vos opérations professionnelles. Pour de plus amples informations sur les fenêtres de maintenance, veuillez consulter AWS Systems Manager Fenêtres de maintenance (p. 688).

Avant de commencer cette procédure, vous devez configurer des rôles et des autorisations pour Fenêtres de maintenance. Pour plus d'informations, consultez la section Configuration d'Fenêtres de maintenance (p. 690).

Pour créer une fenêtre de maintenance pour l'application des correctifs

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fenêtres de maintenance.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Fenêtres de maintenance.
3. Choisissez Create maintenance window (Créer une fenêtre de maintenance).
4. Dans Name (Nom), entrez un nom désignant la fenêtre de maintenance pour l'application des correctifs correspondant aux mises à jour critiques et importantes.
5. Pour Description, entrez une description.
6. Choisissez Allow unregistered targets (Autoriser les cibles non enregistrées) si vous souhaitez autoriser une tâche de fenêtre de maintenance à exécuter des instances gérées, même si vous n'avez pas enregistré ces instances en tant que cibles. Si vous choisissez cette option, vous pouvez choisir des instances non enregistrées (par ID d'instance) lorsque vous enregistrez une tâche avec la fenêtre de maintenance.

   Si vous ne choisissez pas cette option, vous devez choisir des cibles enregistrées au préalable lorsque vous enregistrez une tâche avec la fenêtre de maintenance.
7. En haut de la section Schedule (Planification) spécifiez un programme pour la fenêtre de maintenance à l'aide de l'une des trois options de planification.

Pour plus d'informations sur la génération d'expressions cron/rate, consultez Référence : Expressions cron et rate pour Systems Manager (p. 1392).

8. Pour Duration (Durée), entrez le nombre d'heures pendant lequel la fenêtre de maintenance devra s'exécuter. La valeur que vous spécifiez détermine l'heure de fin de la fenêtre de maintenance en fonction de l'heure de démarrage. Aucune tâche de fenêtre de maintenance n'est autorisée à démarrer après l'heure de fin résultante moins le nombre d'heures que vous spécifiez pour Stop initiating tasks (Arrêt de l'initialisation de tâches) à l'étape suivante.

   Par exemple, si la fenêtre de maintenance commence à 15 h, que la durée est de trois heures et que la valeur de Stop initiating tasks (Arrêt de l'initialisation de tâches) est d'une heure, aucune tâche de fenêtre de maintenance ne peut commencer après 17 h.
9. Dans le champ Stop initiating tasks (Arrêter le lancement des tâches), entrez le nombre d'heures avant la fin de la fenêtre de maintenance pendant lequel le système doit cesser de planifier l'exécution de nouvelles tâches.
10. (Facultatif) Pour Start date (optional) [Date de début (facultatif)], spécifiez la date et l'heure, au format ISO-8601 étendu, où vous voulez que la fenêtre de maintenance devienne active. Cela vous permet de retarder l'activation de la fenêtre de maintenance jusqu'à la date ultérieure spécifiée.
11. (Facultatif) Pour Start date (optional) [Date de début (facultatif)], spécifiez la date et l'heure, au format ISO-8601 étendu, où vous voulez que la fenêtre de maintenance devienne inactive. Cela vous permet de définir une date et une heure futures après lesquelles la fenêtre de maintenance ne s'exécutera plus.
12. (Facultatif) Pour Time zone (optional) [Fuseau horaire (facultatif)], spécifiez le fuseau horaire sur lequel baser les exécutions de fenêtre de maintenance planifiées, au format IANA (Internet Assigned Numbers Authority). Exemple : « Amérique/Los_Angeles », « etc/UTC » ou « Asie/Séoul ».

    Pour plus d'informations sur les formats valides, consultez Time Zone Database sur le site web de l'IANA.
13. Choisissez Create maintenance window (Créer une fenêtre de maintenance).
14. Dans la liste des fenêtres de maintenance, choisissez la fenêtre de maintenance que vous venez de créer, puis choisissez Actions, Register targets (Enregistrer les cibles).
15. (Facultatif) Dans la section Maintenance window target details (Détails de la cible de la fenêtre de maintenance), fournissez un nom, une description et des informations sur le propriétaire (votre nom ou pseudo) pour cette cible.
16. Pour Targets (Cibles), choisissez Specifying instance tags (Spécification des balises d'instance).
17. Pour Instance tags (Balises d'instance), entrez une clé de balise et une valeur de balise pour identifier les instances à enregistrer dans la fenêtre de maintenance, puis choisissez Add (Ajouter).
18. Choisissez Register target (Enregistrer la cible). Le système crée une cible de fenêtre de maintenance.
19. Dans la page des détails de la fenêtre de maintenance que vous avez créée, choisissez Actions, Register run command task (Enregistrer une tâche d'exécution de commande).
20. (Facultatif) Dans Maintenance window task details (Détails de la tâche de fenêtre de maintenance), attribuez un nom et une description à cette tâche.
21. Pour Command document (Document de commande), choisissez AWS-RunPatchBaseline.
22. Pour Task priority (Priorité de tâche), choisissez une priorité. Zéro (`0`) est la priorité la plus élevée.
23. Dans Targets (Cibles), sous Target by (Cible par), choisissez la cible de fenêtre de maintenance que vous avez créée précédemment dans cette procédure.
24. (Facultatif) Pour Contrôle du débit :

    - Pour Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage d'instances sur lesquelles exécuter simultanément la commande.

Note

Si vous avez sélectionné des cibles en indiquant des balises appliquées à des instances gérées ou en indiquant des groupes de ressources AWS, et que vous n'êtes pas sûr du nombre d'instances ciblées, limitez le nombre d'instances pouvant exécuter simultanément le document en spécifiant un pourcentage.

- Pour Seuil d'erreur, spécifiez quand arrêter l'exécution de la commande sur d'autres instances après son échec sur un nombre ou un pourcentage d'instances. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les instances sur lesquelles la commande est en cours de traitement peuvent également envoyer des erreurs.

25. Pour IAM service role (Rôle de service IAM), choisissez l'une des options suivantes pour autoriser Systems Manager à exécuter des tâches sur vos instances cibles :

    - Créer et utiliser un rôle lié à un service pour Systems Manager

      Les rôles liés à un service offrent une manière sécurisée d'accorder des autorisations aux services AWS, car seul le service lié peut assumer un rôle lié à un service. De plus, AWS définit automatiquement les autorisations des rôles liés à un service, en fonction des actions que le service lié effectue en votre nom.

      Note

      Si un rôle lié à un service a déjà été créé pour votre compte, choisissez Use the service-linked role for Systems Manager (Utiliser le rôle lié à un service pour Systems Manager).

    - Use a custom service role (Utiliser un rôle de service personnalisé)

      Vous pouvez créer un rôle de service personnalisé pour les tâches de fenêtre de maintenance si vous souhaitez utiliser des autorisations plus rigoureuses que celles fournies par le rôle lié à un service.

      Si vous avez besoin de créer un rôle de service personnalisé, veuillez consulter l'une des rubriques suivantes :

      - Contrôle de l'accès aux fenêtres de maintenance (console) (p. 692)
      - Contrôler l'accès aux fenêtres de maintenance (AWS CLI) (p. 697)
      - Contrôler l'accès aux fenêtres de maintenance (Outils pour Windows PowerShell) (p. 703)

    Pour choisir plus facilement d'utiliser un rôle de service personnalisé ou le rôle lié à un service Systems Manager avec une tâche de fenêtre de maintenance, consultez Dois-je utiliser un rôle lié à un service ou un rôle de service personnalisé pour exécuter des tâches de fenêtre de maintenance ? (p. 691).

26. (Facultatif) Dans Output options (Options de sortie), pour enregistrer la sortie de la commande dans un fichier, sélectionnez Enable writing to an S3 bucket (Autoriser l'écriture dans un compartiment S3) Entrez les noms de compartiment et de préfixe (dossier) dans les zones.

    Note

    Les autorisations S3 qui accordent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour de plus amples informations, veuillez consulter Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). En outre, si le compartiment S3 spécifié se trouve dans un compte AWS différent, assurez-vous que le profil d'instance associé à l'instance dispose des autorisations nécessaires pour écrire dans ce compartiment.

    Pour diffuser la sortie vers un groupe de CloudWatch Logs journaux , cochez la case CloudWatch de sortie. Tapez le nom du groupe de journaux dans la zone.

27. Dans la section SNS notifications (Notifications SNS), si vous souhaitez que des notifications sur le statut de l'exécution de la commande soient envoyées, cochez la case Enable SNS notifications (Activer les notifications SNS).

    Pour plus d'informations sur la configuration des notifications Amazon SNS pour la fonctionnalité Exécuter la commande, consultez Surveillance des changements de statut Systems Manager à l'aide de notifications Amazon SNS (p. 1344).
28. Pour Parameters (Paramètres) :

    - Pour Operation (Opération), sélectionnez Scan (Analyser) afin de rechercher les correctifs manquants, ou choisissez Install (Installer) pour rechercher et installer les correctifs manquants.
    - Vous n'avez pas besoin de spécifier quoi que ce soit dans le champ Snapshot Id (ID d'instantané). Ce système génère et fournit ce paramètre automatiquement.
    - Vous n'avez pas besoin d'entrer quoi que ce soit dans le champ Liste de remplacement Gestionnaire de correctifs d'installation, sauf si vous souhaitez utiliser un autre ensemble de correctifs que celui spécifié pour la référence de correctifs. Pour plus d'informations, consultez la section Nom du paramètre : `InstallOverrideList` (p. 1083).
    - Pour l'option Redémarrer, spécifiez si vous souhaitez que les instances redémarrent si des correctifs sont installés pendant l'`Install`opération ou si Gestionnaire de correctifs détecte d'autres correctifs installés depuis le dernier redémarrage de l'instance. Pour plus d'informations, consultez la section Nom du paramètre : `RebootOption` (p. 1087).
    - (Facultatif) Pour Comment (Commentaire), entrez une note de suivi ou un rappel concernant cette commande.
    - Pour Timeout (seconds) (Délai (secondes)), entrez le nombre de secondes durant lesquelles le système doit attendre que l'opération se termine avant que celle-ci ne soit considérée comme ayant échoué.
29. Choisissez Register run command task (Enregistrer une tâche d'exécution de commande).

Une fois la tâche de la fenêtre de maintenance terminée, vous pouvez consulter les détails de conformité du correctif dans la console Systems Manager, sur la page Instances gérées. Dans la barre de filtre, utilisez les filtres AWS:PatchSummary et AWS:PatchCompliance.

> **Note**
>
> Vous pouvez enregistrer votre requête en attribuant un signet à l'URL une fois que vous avez spécifié les filtres.

Vous pouvez également consulter une instance spécifique en la choisissant dans la page Instances gérées, puis en choisissant l'onglet Patch (Correctif). Vous pouvez également utiliser et DescribePatchGroupState pour afficher les détails de conformité.DescribeInstancePatchStatesForPatchGroupAPIs Pour plus d'informations sur les données de conformité des correctifs, consultez A propos de la conformité des correctifs (p. 783).

À propos des planifications d'application de correctifs avec Fenêtres de maintenance

Une fois que vous avez configuré une référence de correctifs (et éventuellement un groupe de correctifs), vous pouvez appliquer des correctifs à votre instance à l'aide d'une fenêtre de maintenance. Une fenêtre de maintenance peut réduire l'impact sur la disponibilité du serveur en vous permettant de spécifier une heure de réalisation du processus d'application des correctifs de manière à ne pas interrompre les opérations professionnelles. Une fenêtre de maintenance fonctionne comme suit :

1. Créez une fenêtre de maintenance avec un calendrier pour vos opérations d'application de correctifs.
2. Choisissez les cibles de la fenêtre de maintenance en spécifiant la balise Groupe de correctifs comme nom de balise et n'importe quelle valeur pour laquelle vous avez défini des balises Amazon EC2, par exemple, « serveurs de production ».

3. Créez une nouvelle tâche de fenêtre de maintenance et spécifiez le document AWS-RunPatchBaseline.

Lorsque vous configurez la tâche, vous pouvez choisir d'analyser les instances ou d'analyser et d'installer les correctifs sur les instances. Si vous choisissez d'analyser les instances, Gestionnaire de correctifs analyse chaque instance et génère une liste des correctifs manquants que vous devez passer en revue.

Si vous choisissez d'analyser et d'installer des correctifs, Gestionnaire de correctifs analyse chaque instance et compare la liste des correctifs installés à la liste des correctifs approuvés dans la référence. Gestionnaire de correctifs identifie les correctifs manquants, puis télécharge et installe tous les correctifs manquants et approuvés.

Si vous souhaitez effectuer une analyse unique ou procéder à une installation pour résoudre un problème, vous pouvez utiliser la fonctionnalité Fonctionnalité Exécuter la commande afin d'appeler directement le document AWS-RunPatchBaseline.

> **Important**
>
> Une fois les correctifs installés, Systems Manager redémarre chaque instance. Le redémarrage est nécessaire pour s'assurer que les correctifs sont installés correctement et afin de garantir que le système n'a pas quitté l'instance dans un état potentiellement mauvais. (Exception : si le paramètre `RebootOption` est défini sur `NoReboot` dans le document `AWS-RunPatchBaseline`, l'instance n'est pas redémarrée après l'exécution du Gestionnaire de correctifs. Pour de plus amples informations, veuillez consulter Nom du paramètre : `RebootOption` (p. 1087).)

Intégration d’Gestionnaire de correctifs à AWS Security Hub

AWS Security Hubfournit une vue complète des alertes de sécurité dans AWS . Il regroupe, organise et hiérarchise vos alertes de sécurité, ou conclusions, à partir de plusieurs AWS services et outils partenaires. Cela réduit l'effort de collecte et de hiérarchisation des résultats de sécurité entre les comptes.

Lorsque vous procédez à l'intégration Gestionnaire de correctifs avec Security Hub , Security Hub surveille la posture d'application de correctifs de vos flottes du point de vue de la sécurité. Vous pouvez également exécuter des contrôles automatisés et continus de conformité et de configuration au niveau du compte pour identifier des comptes et des ressources spécifiques nécessitant une attention particulière. Vous pouvez afficher les résultats de sécurité entre les comptes pour voir l'état actuel de la sécurité et de la conformité à l'aide du tableau de bord intégré.

Avec Amazon CloudWatch Events , vous pouvez recevoir des alertes lorsque Security Hub détecte que des instances de votre parc sont non conformes.

Pour de plus amples informations sur Security Hub, veuillez consulter AWS Security Hub Guide de l'utilisateur.

Des frais sont facturés pour l'utilisation d' Security Hub . Pour de plus amples informations, veuillez consulter Tarification de Security Hub.

Activation des AWS Security Hub

Cette procédure s'applique lorsque n'Security Huba pas encore été activé dans le compte.

Prerequisites

Vous devez disposer des autorisations AWS Identity and Access Management requises pour activer IAM .Security Hub Pour plus d'informations, consultez Attachement de la stratégie IAM requise à l'identité IAM dans le Guide AWS Security Hub de l'utilisateur .

Pour activer AWS Security Hub

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.

2. Dans le volet de navigation, choisissez Gestionnaire de correctifs.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Gestionnaire de correctifs.
3. Choisissez l'onglet Settings (Paramètres).
4. Choisissez OpenSecurity Hub.
5. Suivez les étapes décrites dans Activation de Security Hub dans le Guide AWS Security Hub de l'utilisateur .

Ajout Gestionnaire de correctifs de à Security Hub l'intégration

Cette procédure décrit comment intégrer Gestionnaire de correctifs et Security Hub quand Security Hub est déjà actif mais Gestionnaire de correctifs l'intégration est désactivée. Cette procédure ne doit être effectuée que si l'intégration a été désactivée manuellement.

Pour ajouter Gestionnaire de correctifs à lSecurity Hub'intégration

1. Dans le volet de navigation, choisissez Gestionnaire de correctifs.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Gestionnaire de correctifs.
2. Choisissez l'onglet Settings (Paramètres).
3. Sous la Security Hub section Export to (Exporter vers), à droite des résultats de conformité des correctifs ne sont pas exportés vers Security Hub, choisissez Enable (Activer).

Désactivation Gestionnaire de correctifs de l'Security Hubintégration

Cette procédure supprime lGestionnaire de correctifs'intégration à Security Hub . Il ne se désactive pas luiSecurity Hub-même.

Pour désactiver Gestionnaire de correctifs à partir de Security Hub l'intégration

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le volet de navigation, choisissez Gestionnaire de correctifs.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Gestionnaire de correctifs.
3. Choisissez l'onglet Settings (Paramètres).
4. Sous la Security Hub section Export to (Exporter vers), à droite des résultats de conformité des correctifs ne sont pas exportés vers Security Hub, choisissez Disable (Désactiver).

Examen des Security Hub résultats pour la conformité des correctifs

Cette procédure décrit comment afficher les résultats dans Security Hub sur les instances gérées de votre parc qui ne sont pas conformes aux correctifs.

Pour vérifier les Security Hub résultats pour la conformité des correctifs

1. Connectez-vous à AWS Management Console et ouvrez la console AWS Security Hub à l'adresse https://console.aws.amazon.com/securityhub/.
2. Dans le volet de navigation, choisissez Conclusions.
3. Choisissez l'icône Add filters (Ajouter des filtres) (🔍.
4. Dans le menu, sous Filtres, choisissez Nom du produit.
5. Dans la boîte de dialogue qui s'ouvre, choisissez « is » dans le premier champ, puis « enter » **`Systems Manager Gestionnaire de correctifs`** dans le second champ.
6. Choisissez Apply (Appliquer).
7. Ajoutez des filtres supplémentaires pour affiner vos résultats.
8. Dans la liste des résultats, choisissez le titre d'un résultat sur lequel vous souhaitez plus d'informations.

   Un volet s'ouvre sur le côté droit de l'écran avec plus de détails sur la ressource, le problème détecté et une correction recommandée.

   > **Important**
   >
   > A l'heure actuelle, Security Hub indique le type de ressource de toutes les instances gérées en tant que `AwsEc2Instance` . Cela inclut les serveurs et les machines virtuelles sur site que vous avez enregistrés pour une utilisation avec Systems Manager .

Contenu connexe

- Résultats dans leAWS Security Hub Guide de l'utilisateur
- Conformité des correctifs multi-comptes avec et Security Hub Gestionnaire de correctifs dans le Blog AWS consacré à la gestion et la gouvernance

Utilisation dGestionnaire de correctifs' AWS CLI ()

La section inclut des exemples de commandes d'interface de ligne de commande que vous pouvez utiliser pour effectuer des tâches Gestionnaire de correctifs de configuration.

Pour accéder à un exemple d'utilisation de l'AWS CLI afin d'appliquer un correctif à un environnement de serveur à l'aide d'une référence de correctifs personnalisée, consultez Procédure : Appliquer un correctif à un environnement serveur (AWS CLI) (p. 1168).

Pour de plus amples informations sur l'utilisation de l'interface de ligne de commande pour les tâches AWS Systems Manager, veuillez consulter AWS Systems Manager section of the AWS CLI Command Reference.

Exemples de commandes

- Créer une référence de correctifs (p. 1148)
- Création d'une référence de correctifs avec des référentiels personnalisés pour les différentes versions du système d'exploitation (p. 1149)
- Mettre à jour une référence de correctifs (p. 1150)
- Renommer une référence de correctifs (p. 1151)
- Supprimer une référence de correctifs (p. 1152)
- Afficher toutes les références de correctifs (p. 1153)
- Répertorier toutes les références de correctifs fournies par AWS (p. 1154)
- Répertorier mes références de correctifs (p. 1154)

- Afficher une référence de correctifs (p. 1155)
- Obtenir la référence de correctifs par défaut (p. 1156)
- Définir une référence de correctifs personnalisée comme valeur par défaut (p. 1156)
- Réinitialiser une référence de correctifs AWS comme valeur par défaut (p. 1156)
- Créer un groupe de correctifs (p. 1157)
- Baliser une référence de correctifs (p. 1158)
- Enregistrer un groupe de correctifs « Serveurs web » avec un référentiel de correctifs (p. 1158)
- Enregistrer un groupe de correctifs « Serveurs frontaux » dans le référence de correctifs fourni par AWS (p. 1159)
- Afficher les enregistrements du groupe de correctifs (p. 1159)
- Annuler l'enregistrement d'un groupe de correctifs à partir d'une référence de correctifs (p. 1160)
- Obtenir tous les correctifs définis par une référence de correctifs (p. 1160)
- Obtenez tous les correctifs pour AmazonLinux2018 .03 ayant un statut Classification SECURITY et GravitéCRITICAL (p. 1161)
- Obtenir tous les correctifs pour Windows Server 2012 dont la gravité MSRC est Critique (p. 1162)
- Obtenir tous les correctifs disponibles (p. 1162)
- Répertorier les balises d'une référence de correctifs (p. 1164)
- Supprimer une balise d'une référence de correctifs (p. 1164)
- Obtenir le résumé de l'état des correctifs par instance (p. 1164)
- Obtenir les informations de conformité des correctifs pour une instance (p. 1165)

Créer une référence de correctifs

La commande suivante crée une référence de correctifs qui approuve toutes les mises à jour de sécurité critiques et importantes pour Windows Server 2012 R2 cinq jours après leur publication. Des correctifs ont également été spécifiés pour les listes de correctifs approuvés et rejetés. En outre, la référence de correctifs a été balisée pour indiquer qu'elle est destinée à un environnement de production.

Linux

```
aws ssm create-patch-baseline \
    --name "Windows-Server-2012R2" \
    --tags "Key=Environment,Value=Production" \
    --description "Windows Server 2012 R2, Important and Critical security updates" \
    --approved-patches "KB2032276,MS10-048" \
    --rejected-patches "KB2124261" \
    --rejected-patches-action "ALLOW_AS_DEPENDENCY" \
    --approval-rules
 "PatchRules=[{PatchFilterGroup={PatchFilters=[{Key=MSRC_SEVERITY,Values=[Important,Critical]},
{Key=CLASSIFICATION,Values=SecurityUpdates},
{Key=PRODUCT,Values=WindowsServer2012R2}]},ApproveAfterDays=5}]"
```

Windows

```
aws ssm create-patch-baseline ^
    --name "Windows-Server-2012R2" ^
    --tags "Key=Environment,Value=Production" ^
    --description "Windows Server 2012 R2, Important and Critical security updates" ^
    --approved-patches "KB2032276,MS10-048" ^
    --rejected-patches "KB2124261" ^
    --rejected-patches-action "ALLOW_AS_DEPENDENCY" ^
    --approval-rules
 "PatchRules=[{PatchFilterGroup={PatchFilters=[{Key=MSRC_SEVERITY,Values=[Important,Critical]},
```

```
{Key=CLASSIFICATION,Values=SecurityUpdates},
{Key=PRODUCT,Values=WindowsServer2012R2}]},ApproveAfterDays=5}]"
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

Création d'une référence de correctifs avec des référentiels personnalisés pour les différentes versions du système d'exploitation

S'applique uniquement aux instances Linux. La commande suivante montre comment spécifier le référentiel de correctifs à utiliser pour une version spécifique du système d'exploitation Amazon Linux. Cet exemple utilise un référentiel source activé par défaut sur Amazon Linux 2017.09, mais peut être adapté à un autre référentiel source que vous avez configuré pour une instance.

> **Note**
>
> Pour mieux illustrer cette commande plus complexe, nous utilisons l'option `--cli-input-json` avec des options supplémentaires stockées dans un fichier JSON externe.

1. Créez un fichier JSON avec un nom tel que `my-patch-repository.json` et ajoutez-lui le contenu suivant :

```
{
    "Description": "My patch repository for Amazon Linux 2017.09",
    "Name": "Amazon-Linux-2017.09",
    "OperatingSystem": "AMAZON_LINUX",
    "ApprovalRules": {
        "PatchRules": [
            {
                "ApproveAfterDays": 7,
                "EnableNonSecurity": true,
                "PatchFilterGroup": {
                    "PatchFilters": [
                        {
                            "Key": "SEVERITY",
                            "Values": [
                                "Important",
                                "Critical"
                            ]
                        },
                        {
                            "Key": "CLASSIFICATION",
                            "Values": [
                                "Security",
                                "Bugfix"
                            ]
                        },
                        {
                            "Key": "PRODUCT",
                            "Values": [
                                "AmazonLinux2017.09"
                            ]
                        }
                    ]
                }
```

```
            }
        ]
    },
    "Sources": [
        {
            "Name": "My-AL2017.09",
            "Products": [
                "AmazonLinux2017.09"
            ],
            "Configuration": "[amzn-main] \nname=amzn-main-Base\nmirrorlist=http://
repo./$awsregion./$awsdomain//$releasever/main/mirror.list //nmirrorlist_expire=300//
nmetadata_expire=300 \npriority=10 \nfailovermethod=priority \nfastestmirror_enabled=0
 \ngpgcheck=1 \ngpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-amazon-ga \nenabled=1
 \nretries=3 \ntimeout=5\nreport_instanceid=yes"
        }
    ]
}
```

2. Dans le répertoire où vous avez enregistré le fichier, exécutez la commande suivante :

```
aws ssm create-patch-baseline --cli-input-json file://my-patch-repository.json
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
    "BaselineId": "pb-12343b962ba63wxya"
}
```

Mettre à jour une référence de correctifs

La commande suivante ajoute deux correctifs comme refusés et un correctif comme approuvé à une référence de correctifs existante.

> **Note**
>
> Pour plus d'informations sur les formats acceptés pour les listes de correctifs approuvés et de correctifs rejetés, consultez À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés (p. 1103).

Linux

```
aws ssm update-patch-baseline \
    --baseline-id pb-0c10e65780EXAMPLE \
    --rejected-patches "KB2032276" "MS10-048" \
    --approved-patches "KB2124261"
```

Windows

```
aws ssm update-patch-baseline ^
    --baseline-id pb-0c10e65780EXAMPLE ^
    --rejected-patches "KB2032276" "MS10-048" ^
    --approved-patches "KB2124261"
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
```

```
    "BaselineId":"pb-0c10e65780EXAMPLE",
    "Name":"Windows-Server-2012R2",
    "RejectedPatches":[
       "KB2032276",
       "MS10-048"
    ],
    "GlobalFilters":{
       "PatchFilters":[

       ]
    },
    "ApprovalRules":{
       "PatchRules":[
          {
             "PatchFilterGroup":{
                "PatchFilters":[
                   {
                      "Values":[
                         "Important",
                         "Critical"
                      ],
                      "Key":"MSRC_SEVERITY"
                   },
                   {
                      "Values":[
                         "SecurityUpdates"
                      ],
                      "Key":"CLASSIFICATION"
                   },
                   {
                      "Values":[
                         "WindowsServer2012R2"
                      ],
                      "Key":"PRODUCT"
                   }
                ]
             },
             "ApproveAfterDays":5
          }
       ]
    },
    "ModifiedDate":1481001494.035,
    "CreatedDate":1480997823.81,
    "ApprovedPatches":[
       "KB2124261"
    ],
    "Description":"Windows Server 2012 R2, Important and Critical security updates"
}
```

Renommer une référence de correctifs

Linux

```
aws ssm update-patch-baseline \
    --baseline-id pb-0c10e65780EXAMPLE \
    --name "Windows-Server-2012-R2-Important-and-Critical-Security-Updates"
```

Windows

```
aws ssm update-patch-baseline ^
    --baseline-id pb-0c10e65780EXAMPLE ^
    --name "Windows-Server-2012-R2-Important-and-Critical-Security-Updates"
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "BaselineId":"pb-0c10e65780EXAMPLE",
   "Name":"Windows-Server-2012-R2-Important-and-Critical-Security-Updates",
   "RejectedPatches":[
      "KB2032276",
      "MS10-048"
   ],
   "GlobalFilters":{
      "PatchFilters":[

      ]
   },
   "ApprovalRules":{
      "PatchRules":[
         {
            "PatchFilterGroup":{
               "PatchFilters":[
                  {
                     "Values":[
                        "Important",
                        "Critical"
                     ],
                     "Key":"MSRC_SEVERITY"
                  },
                  {
                     "Values":[
                        "SecurityUpdates"
                     ],
                     "Key":"CLASSIFICATION"
                  },
                  {
                     "Values":[
                        "WindowsServer2012R2"
                     ],
                     "Key":"PRODUCT"
                  }
               ]
            },
            "ApproveAfterDays":5
         }
      ]
   },
   "ModifiedDate":1481001795.287,
   "CreatedDate":1480997823.81,
   "ApprovedPatches":[
      "KB2124261"
   ],
   "Description":"Windows Server 2012 R2, Important and Critical security updates"
}
```

Supprimer une référence de correctifs

```
aws ssm delete-patch-baseline --baseline-id "pb-0c10e65780EXAMPLE"
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

Afficher toutes les références de correctifs

```
aws ssm describe-patch-baselines
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "BaselineIdentities":[
      {
         "BaselineName":"AWS-DefaultPatchBaseline",
         "DefaultBaseline":true,
         "BaselineDescription":"Default Patch Baseline Provided by AWS.",
         "BaselineId":"arn:aws:ssm:us-east-2:111122223333:patchbaseline/
pb-0c10e65780EXAMPLE"
      },
      {
         "BaselineName":"Windows-Server-2012R2",
         "DefaultBaseline":false,
         "BaselineDescription":"Windows Server 2012 R2, Important and Critical security
 updates",
         "BaselineId":"pb-0c10e65780EXAMPLE"
      }
   ]
}
```

Voici une autre commande qui répertorie toutes les références de correctifs dans une région.

Linux

```
aws ssm describe-patch-baselines \
    --region us-east-2 \
    --filters "Key=OWNER,Values=[All]"
```

Windows

```
aws ssm describe-patch-baselines ^
    --region us-east-2 ^
    --filters "Key=OWNER,Values=[All]"
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "BaselineIdentities":[
      {
         "BaselineName":"AWS-DefaultPatchBaseline",
         "DefaultBaseline":true,
         "BaselineDescription":"Default Patch Baseline Provided by AWS.",
         "BaselineId":"arn:aws:ssm:us-east-2:111122223333:patchbaseline/
pb-0c10e65780EXAMPLE"
      },
      {
         "BaselineName":"Windows-Server-2012R2",
         "DefaultBaseline":false,
         "BaselineDescription":"Windows Server 2012 R2, Important and Critical security
 updates",
         "BaselineId":"pb-0c10e65780EXAMPLE"
      }
   ]
}
```

Répertorier toutes les références de correctifs fournies par AWS

Linux

```
aws ssm describe-patch-baselines \
    --region us-east-2 \
    --filters "Key=OWNER,Values=[AWS]"
```

Windows

```
aws ssm describe-patch-baselines ^
    --region us-east-2 ^
    --filters "Key=OWNER,Values=[AWS]"
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "BaselineIdentities":[
      {
         "BaselineName":"AWS-DefaultPatchBaseline",
         "DefaultBaseline":true,
         "BaselineDescription":"Default Patch Baseline Provided by AWS.",
         "BaselineId":"arn:aws:ssm:us-east-2:111122223333:patchbaseline/
pb-0c10e65780EXAMPLE"
      }
   ]
}
```

Répertorier mes références de correctifs

Linux

```
aws ssm describe-patch-baselines \
    --region us-east-2 \
    --filters "Key=OWNER,Values=[Self]"
```

Windows

```
aws ssm describe-patch-baselines ^
    --region us-east-2 ^
    --filters "Key=OWNER,Values=[Self]"
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "BaselineIdentities":[
      {
         "BaselineName":"Windows-Server-2012R2",
         "DefaultBaseline":false,
         "BaselineDescription":"Windows Server 2012 R2, Important and Critical security
 updates",
         "BaselineId":"pb-0c10e65780EXAMPLE"
      }
   ]
}
```

Afficher une référence de correctifs

```
aws ssm get-patch-baseline --baseline-id pb-0c10e65780EXAMPLE
```

> **Note**
>
> Pour les références de correctifs personnalisées, vous pouvez spécifier l'ID de référence de correctifs ou l'ARN complet. Pour la référence de correctifs fournie par AWS, vous devez spécifier l'ARN complet. Par exemple, `arn:aws:ssm:us-east-1:075727635805:patchbaseline/pb-03e3f588eec25344c`.

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "BaselineId":"pb-0c10e65780EXAMPLE",
   "Name":"Windows-Server-2012R2",
   "PatchGroups":[
      "Web Servers"
   ],
   "RejectedPatches":[

   ],
   "GlobalFilters":{
      "PatchFilters":[

      ]
   },
   "ApprovalRules":{
      "PatchRules":[
         {
            "PatchFilterGroup":{
               "PatchFilters":[
                  {
                     "Values":[
                        "Important",
                        "Critical"
                     ],
                     "Key":"MSRC_SEVERITY"
                  },
                  {
                     "Values":[
                        "SecurityUpdates"
                     ],
                     "Key":"CLASSIFICATION"
                  },
                  {
                     "Values":[
                        "WindowsServer2012R2"
                     ],
                     "Key":"PRODUCT"
                  }
               ]
            },
            "ApproveAfterDays":5
         }
      ]
   },
   "ModifiedDate":1480997823.81,
   "CreatedDate":1480997823.81,
   "ApprovedPatches":[

   ],
   "Description":"Windows Server 2012 R2, Important and Critical security updates"
```

```
}
```

Obtenir la référence de correctifs par défaut

```
aws ssm get-default-patch-baseline --region us-east-2
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "BaselineId":"arn:aws:ssm:us-east-2:111122223333:patchbaseline/pb-0c10e65780EXAMPLE"
}
```

Définir une référence de correctifs personnalisée comme valeur par défaut

Linux

```
aws ssm register-default-patch-baseline \
    --region us-east-2 \
    --baseline-id "pb-0c10e65780EXAMPLE"
```

Windows

```
aws ssm register-default-patch-baseline ^
    --region us-east-2 ^
    --baseline-id "pb-0c10e65780EXAMPLE"
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
   "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

Réinitialiser une référence de correctifs AWS comme valeur par défaut

Linux

```
aws ssm register-default-patch-baseline \
    --region us-east-2 \
    --baseline-id "arn:aws:ssm:us-east-2:733109147000:patchbaseline/
pb-0574b43a65ea646ed"
```

Windows

```
aws ssm register-default-patch-baseline ^
    --region us-east-2 ^
    --baseline-id "arn:aws:ssm:us-east-2:733109147000:patchbaseline/
pb-0574b43a65ea646ed"
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
   "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

Créer un groupe de correctifs

Pour faciliter l'organisation de votre application de correctifs, nous vous recommandons d'ajouter des instances à des groupes de correctifs en utilisant les balises. Les groupes de correctifs nécessitent l'utilisation de la clé de balise Groupe de correctifs. Vous pouvez spécifier une valeur mais la clé de balise doit être Patch Group (Groupe de correctifs). Pour de plus amples informations sur les groupes de correctifs, veuillez consulter À propos des groupes de correctifs (p. 1105).

Une fois que vous avez regroupé vos instances à l'aide de balises, vous devez ajouter la valeur du groupe de correctifs à une référence de correctifs. En enregistrant le groupe de correctifs auprès d'une référence de correctifs, vous veillez à ce que les correctifs appropriés soient installés lors de l'opération d'application des correctifs.

Tâche 1 : Ajout d'instances EC2 à un groupe de correctifs à l'aide de balises

> **Note**
>
> Lors de l'utilisation de la console Amazon EC2 et de AWS CLI, il est possible d'appliquer des balises `Key = Patch Group` à des instances qui ne sont pas encore configurées pour une utilisation avec Systems Manager. Si une instance EC2 dans laquelle vous vous attendez à voir Gestionnaire de correctifs n'est pas répertoriée après l'application de la `Patch Group` balise, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.

Exécutez la commande suivante pour ajouter la balise `Patch Group` à une instance EC2.

```
aws ec2 create-tags --resources "i-1234567890abcdef0" --tags "Key=Patch
 Group,Value=GroupValue"
```

Tâche 2 : Ajouter des instances gérées à un groupe de correctifs à l'aide de balises

Exécutez la commande suivante pour ajouter la balise `Patch Group` à une instance gérée.

Linux

```
aws ssm add-tags-to-resource \
    --resource-type "ManagedInstance" \
    --resource-id "mi-0123456789abcdefg" \
    --tags "Key=Patch Group,Value=GroupValue"
```

Windows

```
aws ssm add-tags-to-resource ^
    --resource-type "ManagedInstance" ^
    --resource-id "mi-0123456789abcdefg" ^
    --tags "Key=Patch Group,Value=GroupValue"
```

Tâche 3 : Ajout d'un groupe de correctifs à une référence de correctifs

Exécutez la commande suivante pour associer une valeur de balise `Patch Group` à la référence de correctifs spécifiée.

Linux

```
aws ssm register-patch-baseline-for-patch-group \
    --baseline-id "pb-0123456789abcdef0" \
    --patch-group "Development"
```

Windows

```
aws ssm register-patch-baseline-for-patch-group ^
    --baseline-id "pb-0123456789abcdef0" ^
    --patch-group "Development"
```

Le système renvoie des informations similaires à celles qui suivent :

```
{
  "PatchGroup": "Development",
  "BaselineId": "pb-0123456789abcdef0"
}
```

Baliser une référence de correctifs

Linux

```
aws ssm add-tags-to-resource \
    --resource-type "PatchBaseline" \
    --resource-id "pb-0c10e65780EXAMPLE" \
    --tags "Key=Project,Value=Testing"
```

Windows

```
aws ssm add-tags-to-resource ^
    --resource-type "PatchBaseline" ^
    --resource-id "pb-0c10e65780EXAMPLE" ^
    --tags "Key=Project,Value=Testing"
```

Enregistrer un groupe de correctifs « Serveurs web » avec un référentiel de correctifs

Linux

```
aws ssm register-patch-baseline-for-patch-group \
    --baseline-id "pb-0c10e65780EXAMPLE" \
    --patch-group "Web Servers"
```

Windows

```
aws ssm register-patch-baseline-for-patch-group ^
    --baseline-id "pb-0c10e65780EXAMPLE" ^
    --patch-group "Web Servers"
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "PatchGroup":"Web Servers",
   "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

Enregistrer un groupe de correctifs « Serveurs frontaux » dans le référence de correctifs fourni par AWS

Linux

```
aws ssm register-patch-baseline-for-patch-group \
    --region us-east-2 \
    --baseline-id "arn:aws:ssm:us-east-2:111122223333:patchbaseline/
pb-0c10e65780EXAMPLE" \
    --patch-group "Backend"
```

Windows

```
aws ssm register-patch-baseline-for-patch-group ^
    --region us-east-2 ^
    --baseline-id "arn:aws:ssm:us-east-2:111122223333:patchbaseline/
pb-0c10e65780EXAMPLE" ^
    --patch-group "Backend"
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "PatchGroup":"Backend",
   "BaselineId":"arn:aws:ssm:us-east-2:111122223333:patchbaseline/pb-0c10e65780EXAMPLE"
}
```

Afficher les enregistrements du groupe de correctifs

```
aws ssm describe-patch-groups --region us-east-2
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "PatchGroupPatchBaselineMappings":[
      {
         "PatchGroup":"Backend",
         "BaselineIdentity":{
            "BaselineName":"AWS-DefaultPatchBaseline",
            "DefaultBaseline":false,
            "BaselineDescription":"Default Patch Baseline Provided by AWS.",
            "BaselineId":"arn:aws:ssm:us-east-2:111122223333:patchbaseline/
pb-0c10e65780EXAMPLE"
         }
      },
      {
         "PatchGroup":"Web Servers",
         "BaselineIdentity":{
            "BaselineName":"Windows-Server-2012R2",
            "DefaultBaseline":true,
            "BaselineDescription":"Windows Server 2012 R2, Important and Critical updates",
            "BaselineId":"pb-0c10e65780EXAMPLE"
```

```
            }
        }
    ]
}
```

Annuler l'enregistrement d'un groupe de correctifs à partir d'une référence de correctifs

Linux

```
aws ssm deregister-patch-baseline-for-patch-group \
    --region us-east-2 \
    --patch-group "Production" \
    --baseline-id "arn:aws:ssm:us-east-2:111122223333:patchbaseline/
pb-0c10e65780EXAMPLE"
```

Windows

```
aws ssm deregister-patch-baseline-for-patch-group ^
    --region us-east-2 ^
    --patch-group "Production" ^
    --baseline-id "arn:aws:ssm:us-east-2:111122223333:patchbaseline/
pb-0c10e65780EXAMPLE"
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "PatchGroup":"Production",
   "BaselineId":"arn:aws:ssm:us-east-2:111122223333:patchbaseline/pb-0c10e65780EXAMPLE"
}
```

Obtenir tous les correctifs définis par une référence de correctifs

> **Note**
>
> Cette commande est prise en charge uniquement pour les références de Windows Server correctifs.

Linux

```
aws ssm describe-effective-patches-for-patch-baseline \
    --region us-east-2 \
    --baseline-id "pb-0c10e65780EXAMPLE"
```

Windows

```
aws ssm describe-effective-patches-for-patch-baseline ^
    --region us-east-2 ^
    --baseline-id "pb-0c10e65780EXAMPLE"
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "NextToken":"--token string truncated--",
```

```
    "EffectivePatches":[
       {
          "PatchStatus":{
             "ApprovalDate":1384711200.0,
             "DeploymentStatus":"APPROVED"
          },
          "Patch":{
             "ContentUrl":"https://support.microsoft.com/en-us/kb/2876331",
             "ProductFamily":"Windows",
             "Product":"WindowsServer2012R2",
             "Vendor":"Microsoft",
             "Description":"A security issue has been identified in a Microsoft software
                product that could affect your system. You can help protect your system
                by installing this update from Microsoft. For a complete listing of the
                issues that are included in this update, see the associated Microsoft
                Knowledge Base article. After you install this update, you may have to
                restart your system.",
             "Classification":"SecurityUpdates",
             "Title":"Security Update for Windows Server 2012 R2 Preview (KB2876331)",
             "ReleaseDate":1384279200.0,
             "MsrcClassification":"Critical",
             "Language":"All",
             "KbNumber":"KB2876331",
             "MsrcNumber":"MS13-089",
             "Id":"e74ccc76-85f0-4881-a738-59e9fc9a336d"
          }
       },
       {
          "PatchStatus":{
             "ApprovalDate":1428858000.0,
             "DeploymentStatus":"APPROVED"
          },
          "Patch":{
             "ContentUrl":"https://support.microsoft.com/en-us/kb/2919355",
             "ProductFamily":"Windows",
             "Product":"WindowsServer2012R2",
             "Vendor":"Microsoft",
             "Description":"Windows Server 2012 R2 Update is a cumulative
                set of security updates, critical updates and updates. You
                must install Windows Server 2012 R2 Update to ensure that
                your computer can continue to receive future Windows Updates,
                including security updates. For a complete listing of the
                issues that are included in this update, see the associated
                Microsoft Knowledge Base article for more information. After
                you install this item, you may have to restart your computer.",
             "Classification":"SecurityUpdates",
             "Title":"Windows Server 2012 R2 Update (KB2919355)",
             "ReleaseDate":1428426000.0,
             "MsrcClassification":"Critical",
             "Language":"All",
             "KbNumber":"KB2919355",
             "MsrcNumber":"MS14-018",
             "Id":"8452bac0-bf53-4fbd-915d-499de08c338b"
          }
       }
      ---output truncated---
```

Obtenez tous les correctifs pour AmazonLinux2018 .03 ayant un statut Classification `SECURITY` et Gravité`CRITICAL`

Linux

```
aws ssm describe-available-patches \
```

```
    --region us-east-2 \
    --filters Key=PRODUCT,Values=AmazonLinux2018.03 Key=SEVERITY,Values=CRITCAL
```

Windows

```
aws ssm describe-available-patches ^
    --region us-east-2 ^
    --filters Key=PRODUCT,Values=AmazonLinux2018.03 Key=SEVERITY,Values=CRITCAL
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
    "Patches": [
        {
            "AdvisoryIds": ["ALAS-2011-1"],
            "BugzillaIds": [ "1234567" ],
            "Classification": "SECURITY",
            "CVEIds": [ "CVE-2011-3192"],
            "Name": "zziplib",
            "Epoch": "0",
            "Version": "2.71",
            "Release": "1.3.amzn1",
            "Arch": "i686",
            "Product": "AmazonLinux2018.03",
            "ReleaseDate": 1590519815,
            "Severity": "CRITICAL"
        }
    ]
}
---output truncated---
```

Obtenir tous les correctifs pour Windows Server 2012 dont la gravité MSRC est Critique

Linux

```
aws ssm describe-available-patches \
    --region us-east-2 \
    --filters Key=PRODUCT,Values=WindowsServer2012 Key=MSRC_SEVERITY,Values=Critical
```

Windows

```
aws ssm describe-available-patches ^
    --region us-east-2 ^
    --filters Key=PRODUCT,Values=WindowsServer2012 Key=MSRC_SEVERITY,Values=Critical
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "Patches":[
      {
         "ContentUrl":"https://support.microsoft.com/en-us/kb/2727528",
         "ProductFamily":"Windows",
         "Product":"WindowsServer2012",
         "Vendor":"Microsoft",
         "Description":"A security issue has been identified that could
```

```
            allow an unauthenticated remote attacker to compromise your
            system and gain control over it. You can help protect your
            system by installing this update from Microsoft. After you
            install this update, you may have to restart your system.",
         "Classification":"SecurityUpdates",
         "Title":"Security Update for Windows Server 2012 (KB2727528)",
         "ReleaseDate":1352829600.0,
         "MsrcClassification":"Critical",
         "Language":"All",
         "KbNumber":"KB2727528",
         "MsrcNumber":"MS12-072",
         "Id":"1eb507be-2040-4eeb-803d-abc55700b715"
      },
      {
         "ContentUrl":"https://support.microsoft.com/en-us/kb/2729462",
         "ProductFamily":"Windows",
         "Product":"WindowsServer2012",
         "Vendor":"Microsoft",
         "Description":"A security issue has been identified that could
            allow an unauthenticated remote attacker to compromise your
            system and gain control over it. You can help protect your
            system by installing this update from Microsoft. After you
            install this update, you may have to restart your system.",
         "Classification":"SecurityUpdates",
         "Title":"Security Update for Microsoft .NET Framework 3.5 on
            Windows 8 and Windows Server 2012 for x64-based Systems (KB2729462)",
         "ReleaseDate":1352829600.0,
         "MsrcClassification":"Critical",
         "Language":"All",
         "KbNumber":"KB2729462",
         "MsrcNumber":"MS12-074",
         "Id":"af873760-c97c-4088-ab7e-5219e120eab4"
      }

---output truncated---
```

Obtenir tous les correctifs disponibles

```
aws ssm describe-available-patches --region us-east-2
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "NextToken":"--token string truncated--",
   "Patches":[
      {
         "ContentUrl":"https://support.microsoft.com/en-us/kb/2032276",
         "ProductFamily":"Windows",
         "Product":"WindowsServer2008R2",
         "Vendor":"Microsoft",
         "Description":"A security issue has been identified that could allow an
            unauthenticated remote attacker to compromise your system and gain
            control over it. You can help protect your system by installing this
            update from Microsoft. After you install this update, you may have to
            restart your system.",
         "Classification":"SecurityUpdates",
         "Title":"Security Update for Windows Server 2008 R2 x64 Edition (KB2032276)",
         "ReleaseDate":1279040400.0,
         "MsrcClassification":"Important",
         "Language":"All",
         "KbNumber":"KB2032276",
         "MsrcNumber":"MS10-043",
         "Id":"8692029b-a3a2-4a87-a73b-8ea881b4b4d6"
```

```
    },
    {
       "ContentUrl":"https://support.microsoft.com/en-us/kb/2124261",
       "ProductFamily":"Windows",
       "Product":"Windows7",
       "Vendor":"Microsoft",
       "Description":"A security issue has been identified that could allow
         an unauthenticated remote attacker to compromise your system and gain
         control over it. You can help protect your system by installing this
         update from Microsoft. After you install this update, you may have
         to restart your system.",
       "Classification":"SecurityUpdates",
       "Title":"Security Update for Windows 7 (KB2124261)",
       "ReleaseDate":1284483600.0,
       "MsrcClassification":"Important",
       "Language":"All",
       "KbNumber":"KB2124261",
       "MsrcNumber":"MS10-065",
       "Id":"12ef1bed-0dd2-4633-b3ac-60888aa8ba33"
    }
    ---output truncated---
```

Répertorier les balises d'une référence de correctifs

Linux

```
aws ssm list-tags-for-resource \
    --resource-type "PatchBaseline" \
    --resource-id "pb-0c10e65780EXAMPLE"
```

Windows

```
aws ssm list-tags-for-resource ^
    --resource-type "PatchBaseline" ^
    --resource-id "pb-0c10e65780EXAMPLE"
```

Supprimer une balise d'une référence de correctifs

Linux

```
aws ssm remove-tags-from-resource \
    --resource-type "PatchBaseline" \
    --resource-id "pb-0c10e65780EXAMPLE" \
    --tag-keys "Project"
```

Windows

```
aws ssm remove-tags-from-resource ^
    --resource-type "PatchBaseline" ^
    --resource-id "pb-0c10e65780EXAMPLE" ^
    --tag-keys "Project"
```

Obtenir le résumé de l'état des correctifs par instance

Le résumé par instance vous fournit un certain nombre de correctifs dans les états suivants par instance : « NotApplicable », « Manquant », « Échec », « InstalledOther » et « Installé ».

Linux

```
aws ssm describe-instance-patch-states \
    --instance-ids i-08ee91c0b17045407 i-09a618aec652973a9
```

Windows

```
aws ssm describe-instance-patch-states ^
    --instance-ids i-08ee91c0b17045407 i-09a618aec652973a9
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "InstancePatchStates":[
      {
            "InstanceId": "i-08ee91c0b17045407",
            "PatchGroup": "",
            "BaselineId": "pb-0e392de35e7c563b7",
            "SnapshotId": "6d03d6c5-f79d-41d0-8d0e-00a9aEXAMPLE",
            "InstalledCount": 50,
            "InstalledOtherCount": 353,
            "InstalledPendingRebootCount": 0,
            "InstalledRejectedCount": 0,
            "MissingCount": 0,
            "FailedCount": 0,
            "UnreportedNotApplicableCount": -1,
            "NotApplicableCount": 671,
            "OperationStartTime": "2020-01-24T12:37:56-08:00",
            "OperationEndTime": "2020-01-24T12:37:59-08:00",
            "Operation": "Scan",
            "RebootOption": "NoReboot"
        },
        {
            "InstanceId": "i-09a618aec652973a9",
            "PatchGroup": "",
            "BaselineId": "pb-07e6d4e9bc703f2e3",
            "SnapshotId": "c7e0441b-1eae-411b-8aa7-973e6EXAMPLE",
            "InstalledCount": 36,
            "InstalledOtherCount": 396,
            "InstalledPendingRebootCount": 0,
            "InstalledRejectedCount": 0,
            "MissingCount": 3,
            "FailedCount": 0,
            "UnreportedNotApplicableCount": -1,
            "NotApplicableCount": 420,
            "OperationStartTime": "2020-01-24T12:37:34-08:00",
            "OperationEndTime": "2020-01-24T12:37:37-08:00",
            "Operation": "Scan",
            "RebootOption": "NoReboot"
        }
     ---output truncated---
```

Obtenir les informations de conformité des correctifs pour une instance

```
aws ssm describe-instance-patches --instance-id i-08ee91c0b17045407
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "NextToken":"--token string truncated--",
   "Patches":[
      {
            "Title": "bind-libs.x86_64:32:9.8.2-0.68.rc1.60.amzn1",
            "KBId": "bind-libs.x86_64",
            "Classification": "Security",
            "Severity": "Important",
            "State": "Installed",
            "InstalledTime": "2019-08-26T11:05:24-07:00"
        },
        {
            "Title": "bind-utils.x86_64:32:9.8.2-0.68.rc1.60.amzn1",
            "KBId": "bind-utils.x86_64",
            "Classification": "Security",
            "Severity": "Important",
            "State": "Installed",
            "InstalledTime": "2019-08-26T11:05:32-07:00"
        },
        {
            "Title": "dhclient.x86_64:12:4.1.1-53.P1.28.amzn1",
            "KBId": "dhclient.x86_64",
            "Classification": "Security",
            "Severity": "Important",
            "State": "Installed",
            "InstalledTime": "2019-08-26T11:05:31-07:00"
        },
    ---output truncated---
```

Procédures Gestionnaire de correctifs AWS Systems Manager

Les procédures de cette section montrent comment utiliser Gestionnaire de correctifs pour une sélection de scénarios d'application de correctifs.

Rubriques

- Procédure : Créer une référence de correctifs pour l'installation des Service Packs Windows (console) (p. 1166)
- Procédure : Appliquer un correctif à un environnement serveur (AWS CLI) (p. 1168)

Procédure : Créer une référence de correctifs pour l'installation des Service Packs Windows (console)

Lorsque vous créez une référence de correctifs personnalisée, vous pouvez spécifier que tous, certains ou un seul type de correctif pris en charge sont installés.

Dans les références de correctifs pour Windows, vous pouvez sélectionner `ServicePacks` comme seule option de Classification afin de limiter les mises à jour des correctifs aux Service Packs uniquement. Les Service Packs peuvent être installés automatiquement à Gestionnaire de correctifs condition que la mise à jour soit disponible dans Windows Update ou Windows Server Update Services (WSUS).

Vous pouvez configurer une référence de correctifs pour contrôler si les Service Packs pour toutes les versions de Windows sont installés, ou uniquement ceux pour des versions spécifiques, comme Windows 7 ou Windows Server 2016.

Suivez la procédure suivante pour créer une référence de correctifs personnalisée qui sera utilisée exclusivement pour l'installation de tous les Service Packs sur vos instances Windows.

Pour utiliser Gestionnaire de correctifs pour installer les Service Packs Windows (console) (Windows)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le volet de navigation, choisissez Gestionnaire de correctifs.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Gestionnaire de correctifs.
3. Choisissez Créer une référence de correctif.
4. Pour Name (Nom), entrez un nom pour votre nouvelle référence de correctifs, par exemple : **`MyWindowsServicePackPatchBaseline`**.
5. (Facultatif) Pour Description, saisissez une description pour cette référence de correctif.
6. Pour Operating system (Système d'exploitation), choisissez `Windows`.
7. Si vous souhaitez commencer à utiliser cette référence de correctif comme valeur par défaut pour Windows dès sa création, sélectionnez Définir cette référence de correctif comme référence par défaut pour les instances Windows Server.

   Si vous choisissez de ne pas définir cette référence de correctifs pour une utilisation immédiate, vous pouvez le faire plus tard. Pour plus d'informations, consultez la section Définition d'une référence de correctifs existante comme référence par défaut (p. 1134).
8. Dans la section Règles d'approbation pour les systèmes d'exploitation), utilisez les champs pour créer une ou plusieurs règles d'approbation automatique.

   - Produit : Versions du système d'exploitation auxquelles la règle d'approbation s'applique, telles que `WindowsServer2008` . Vous pouvez choisir une, plusieurs ou toutes les versions prises en charge de Windows. La sélection par défaut est `All`.
   - Classification : Choisissez `ServicePacks` .
   - Gravité : La valeur de gravité des correctifs à laquelle la règle va s'appliquer. Pour vous assurer que tous les Service Packs sont inclus dans la règle, choisissez `All`.
   - Approbation automatique : Méthode de sélection des correctifs pour approbation automatique.
     - Approuvez les correctifs après un nombre de jours spécifié : Nombre de jours qu' doit attendre après la publication d'un correctif avant quGestionnaire de correctifs'un correctif soit automatiquement approuvé. Vous pouvez entrer tout nombre entier situé entre zéro (0) et 100.
     - Approuvez les correctifs publiés jusqu'à une date spécifique : Date de publication des correctifs pour laquelle applique Gestionnaire de correctifs automatiquement tous les correctifs publiés à cette date ou avant. Par exemple, si vous spécifiez le 7 juillet 2020, aucun correctif publié à partir du 8 juillet 2020 n'est installé automatiquement.
   - (Facultatif) Rapport de conformité : Niveau de gravité que vous voulez attribuer aux Service Packs approuvés par la référence, tel que `High` .

     Note

     Si un Service Pack approuvé est signalé comme manquant, l'option que vous choisissez dans Rapport de conformité, comme `Critical` ou `Medium`, détermine l'importance de la violation de conformité.
9. (Facultatif) Pour Gérer les balises, appliquez une ou plusieurs paires nom/valeur de clé de balise à la référence de correctifs.

   Les balises sont des métadonnées facultatives que vous affectez à une ressource. Les balises vous permettent de classer une ressource de différentes façons, par exemple, par objectif, par propriétaire ou par environnement. Pour cette référence de correctifs dédiée à la mise à jour des Service Packs, vous pouvez spécifier des paires clé-valeur telles que les suivantes :

- `Key=OS,Value=Windows`
- `Key=Classification,Value=ServicePacks`

10. Choisissez Créer une référence de correctif.

Procédure : Appliquer un correctif à un environnement serveur (AWS CLI)

La procédure suivante décrit l'application de correctifs à un environnement de serveur à l'aide d'une référence de correctifs personnalisée, de groupes de correctifs et d'une fenêtre de maintenance.

Avant de commencer

- Installez ou mettez à jour l'Agent SSM sur vos instances. Pour corriger les instances Linux, vos instances doivent exécuter l'Agent SSM version 2.0.834.0 ou ultérieure. Pour plus d'informations, consultez la section Mettre à jour l'Agent SSM à l'aide de Fonctionnalité Exécuter la commande (p. 948).
- Configurez les rôles et les autorisations pour la fonctionnalité Fenêtres de maintenance. Pour plus d'informations, consultez la section Configuration d'Fenêtres de maintenance (p. 690).
- Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

  Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

Pour configurer Gestionnaire de correctifs et corriger les instances (ligne de commande)

1. Exécutez la commande suivante pour créer une référence de correctifs pour Windows nommée `Production-Baseline` . Cette référence de correctifs approuve les correctifs pour un environnement de production sept jours après leur publication. Cela signifie que la référence de correctifs a été balisée pour indiquer qu'elle est destinée à un environnement de production.

   > **Note**
   >
   > Le paramètre `OperatingSystem` et `PatchFilters` varient en fonction du système d'exploitation des instances auxquelles s'applique la référence de correctifs. Pour de plus amples informations, veuillez consulter OperatingSystem et PatchFilter.

   Linux

   ```
   aws ssm create-patch-baseline \
       --name "Production-Baseline" \
       --operating-system "WINDOWS" \
       --tags "Key=Environment,Value=Production" \
       --approval-rules
    "PatchRules=[{PatchFilterGroup={PatchFilters=[{Key=MSRC_SEVERITY,Values=[Critical,Important]},
   {Key=CLASSIFICATION,Values=[SecurityUpdates,Updates,ServicePacks,UpdateRollups,CriticalUpdates]
    \
       --description "Baseline containing all updates approved for production systems"
   ```

   Windows

   ```
   aws ssm create-patch-baseline ^
       --name "Production-Baseline" ^
       --operating-system "WINDOWS" ^
       --tags "Key=Environment,Value=Production" ^
       --approval-rules
    "PatchRules=[{PatchFilterGroup={PatchFilters=[{Key=MSRC_SEVERITY,Values=[Critical,Important]},
   ```

```
{Key=CLASSIFICATION,Values=[SecurityUpdates,Updates,ServicePacks,UpdateRollups,CriticalUpdates]
 ^
    --description "Baseline containing all updates approved for production systems"
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

2. Exécutez les commandes suivantes pour enregistrer la référence de correctifs « Production-Baseline » pour deux groupes de correctifs. Les groupes sont nommés « serveurs de base de données » et « serveurs front-end ».

   Linux

   ```
   aws ssm register-patch-baseline-for-patch-group \
       --baseline-id pb-0c10e65780EXAMPLE \
       --patch-group "Database Servers"
   ```

   Windows

   ```
   aws ssm register-patch-baseline-for-patch-group ^
       --baseline-id pb-0c10e65780EXAMPLE ^
       --patch-group "Database Servers"
   ```

   Le système renvoie des informations similaires à celles qui suivent.

   ```
   {
      "PatchGroup":"Database Servers",
      "BaselineId":"pb-0c10e65780EXAMPLE"
   }
   ```

   Linux

   ```
   aws ssm register-patch-baseline-for-patch-group \
       --baseline-id pb-0c10e65780EXAMPLE \
       --patch-group "Front-End Servers"
   ```

   Windows

   ```
   aws ssm register-patch-baseline-for-patch-group ^
       --baseline-id pb-0c10e65780EXAMPLE ^
       --patch-group "Front-End Servers"
   ```

   Le système renvoie des informations similaires à celles qui suivent.

   ```
   {
      "PatchGroup":"Front-End Servers",
      "BaselineId":"pb-0c10e65780EXAMPLE"
   }
   ```

3. Exécutez les commandes suivantes pour créer deux fenêtres de maintenance pour les serveurs de production. La première fenêtre est exécutée tous les mardis à 22:00. La seconde fenêtre est exécutée

tous les samedis à 22:00. En outre, la fenêtre de maintenance a été balisée pour indiquer qu'elle est destinée à un environnement de production.

Linux

```
aws ssm create-maintenance-window \
    --name "Production-Tuesdays" \
    --tags "Key=Environment,Value=Production" \
    --schedule "cron(0 0 22 ? * TUE *)" \
    --duration 1 \
    --cutoff 0 \
    --no-allow-unassociated-targets
```

Windows

```
aws ssm create-maintenance-window ^
    --name "Production-Tuesdays" ^
    --tags "Key=Environment,Value=Production" ^
    --schedule "cron(0 0 22 ? * TUE *)" ^
    --duration 1 ^
    --cutoff 0 ^
    --no-allow-unassociated-targets
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "WindowId":"mw-0c50858d01EXAMPLE"
}
```

Linux

```
aws ssm create-maintenance-window \
    --name "Production-Saturdays" \
    --tags "Key=Environment,Value=Production" \
    --schedule "cron(0 0 22 ? * SAT *)" \
    --duration 2 \
    --cutoff 0 \
    --no-allow-unassociated-targets
```

Windows

```
aws ssm create-maintenance-window ^
    --name "Production-Saturdays" ^
    --tags "Key=Environment,Value=Production" ^
    --schedule "cron(0 0 22 ? * SAT *)" ^
    --duration 2 ^
    --cutoff 0 ^
    --no-allow-unassociated-targets
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "WindowId":"mw-9a8b7c6d5eEXAMPLE"
}
```

4. Exécutez les commandes suivantes pour enregistrer les groupes de correctifs des serveurs `Database` et `Front-End` avec leurs fenêtres de maintenance respectives.

   Linux

   ```
   aws ssm register-target-with-maintenance-window \
       --window-id mw-0c50858d01EXAMPLE \
       --targets "Key=tag:Patch Group,Values=Database Servers" \
       --owner-information "Database Servers" \
       --resource-type "INSTANCE"
   ```

   Windows

   ```
   aws ssm register-target-with-maintenance-window ^
       --window-id mw-0c50858d01EXAMPLE ^
       --targets "Key=tag:Patch Group,Values=Database Servers" ^
       --owner-information "Database Servers" ^
       --resource-type "INSTANCE"
   ```

   Le système renvoie des informations similaires à celles qui suivent.

   ```
   {
      "WindowTargetId":"e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
   }
   ```

   Linux

   ```
   aws ssm register-target-with-maintenance-window \
   --window-id mw-9a8b7c6d5eEXAMPLE \
   --targets "Key=tag:Patch Group,Values=Front-End Servers" \
   --owner-information "Front-End Servers" \
   --resource-type "INSTANCE"
   ```

   Windows

   ```
   aws ssm register-target-with-maintenance-window ^
       --window-id mw-9a8b7c6d5eEXAMPLE ^
       --targets "Key=tag:Patch Group,Values=Front-End Servers" ^
       --owner-information "Front-End Servers" ^
       --resource-type "INSTANCE"
   ```

   Le système renvoie des informations similaires à celles qui suivent.

   ```
   {
      "WindowTargetId":"faa01c41-1d57-496c-ba77-ff9caEXAMPLE"
   }
   ```

5. Exécutez les commandes suivantes pour enregistrer une tâche de correctif qui installe les mises à jour manquantes sur les serveurs `Database` et `Front-End` pendant leurs fenêtres de maintenance respectives.

   Linux

   ```
   aws ssm register-task-with-maintenance-window \
       --window-id mw-0c50858d01EXAMPLE \
   ```

```
    --targets "Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE" \
    --task-arn "AWS-RunPatchBaseline" \
    --service-role-arn "arn:aws:iam::12345678:role/MW-Role" \
    --task-type "RUN_COMMAND" \
    --max-concurrency 2 \
    --max-errors 1 \
    --priority 1 \
    --task-invocation-parameters "RunCommand={Parameters={Operation=Install}}"
```

Windows

```
aws ssm register-task-with-maintenance-window ^
    --window-id mw-0c50858d01EXAMPLE ^
    --targets "Key=WindowTargetIds,Values=e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE" ^
    --task-arn "AWS-RunPatchBaseline" ^
    --service-role-arn "arn:aws:iam::12345678:role/MW-Role" ^
    --task-type "RUN_COMMAND" ^
    --max-concurrency 2 ^
    --max-errors 1 ^
    --priority 1 ^
    --task-invocation-parameters "RunCommand={Parameters={Operation=Install}}"
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
   "WindowTaskId":"4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE"
}
```

Linux

```
aws ssm register-task-with-maintenance-window \
    --window-id mw-9a8b7c6d5eEXAMPLE \
    --targets "Key=WindowTargetIds,Values=faa01c41-1d57-496c-ba77-ff9caEXAMPLE" \
    --task-arn "AWS-RunPatchBaseline" \
    --service-role-arn "arn:aws:iam::12345678:role/MW-Role" \
    --task-type "RUN_COMMAND" \
    --max-concurrency 2 \
    --max-errors 1 \
    --priority 1 \
    --task-invocation-parameters "RunCommand={Parameters={Operation=Install}}"
```

Windows

```
aws ssm register-task-with-maintenance-window ^
    --window-id mw-9a8b7c6d5eEXAMPLE ^
    --targets "Key=WindowTargetIds,Values=faa01c41-1d57-496c-ba77-ff9caEXAMPLE" ^
    --task-arn "AWS-RunPatchBaseline" ^
    --service-role-arn "arn:aws:iam::12345678:role/MW-Role" ^
    --task-type "RUN_COMMAND" ^
    --max-concurrency 2 ^
    --max-errors 1 ^
    --priority 1 ^
    --task-invocation-parameters "RunCommand={Parameters={Operation=Install}}"
```

Le système renvoie des informations similaires à celles qui suivent.

```
{
```

```
    "WindowTaskId":"8a5c4629-31b0-4edd-8aea-33698EXAMPLE"
}
```

6. Exécutez la commande suivante pour obtenir le résumé de haut niveau de la conformité des correctifs d'un groupe de correctifs. Le résumé de la conformité des correctifs de haut niveau inclut le nombre d'instances avec des correctifs dans les états respectifs des correctifs.

   Note

   Vous devez théoriquement voir des zéros pour le nombre d'instances dans le récapitulatif jusqu'à ce que la tâche de correctif soit exécutée pendant la première fenêtre de maintenance.

   Linux

```
aws ssm describe-patch-group-state \
    --patch-group "Database Servers"
```

   Windows

```
aws ssm describe-patch-group-state ^
    --patch-group "Database Servers"
```

   Le système renvoie des informations similaires à celles qui suivent.

```
{
    "Instances": number,
    "InstancesWithFailedPatches": number,
    "InstancesWithInstalledOtherPatches": number,
    "InstancesWithInstalledPatches": number,
    "InstancesWithInstalledPendingRebootPatches": number,
    "InstancesWithInstalledRejectedPatches": number,
    "InstancesWithMissingPatches": number,
    "InstancesWithNotApplicablePatches": number,
    "InstancesWithUnreportedNotApplicablePatches": number
}
```

7. Exécutez la commande suivante pour obtenir le résumé des états des correctifs par instance pour un groupe de correctifs. Le récapitulatif par instance inclut un certain nombre de correctifs dans les états respectifs des correctifs par instance pour un groupe de correctifs.

   Linux

```
aws ssm describe-instance-patch-states-for-patch-group \
    --patch-group "Database Servers"
```

   Windows

```
aws ssm describe-instance-patch-states-for-patch-group ^
    --patch-group "Database Servers"
```

   Le système renvoie des informations similaires à celles qui suivent.

```
{
    "InstancePatchStates": [
        {
```

```
            "BaselineId": "string",
            "FailedCount": number,
            "InstalledCount": number,
            "InstalledOtherCount": number,
            "InstalledPendingRebootCount": number,
            "InstalledRejectedCount": number,
            "InstallOverrideList": "string",
            "InstanceId": "string",
            "LastNoRebootInstallOperationTime": number,
            "MissingCount": number,
            "NotApplicableCount": number,
            "Operation": "string",
            "OperationEndTime": number,
            "OperationStartTime": number,
            "OwnerInformation": "string",
            "PatchGroup": "string",
            "RebootOption": "string",
            "SnapshotId": "string",
            "UnreportedNotApplicableCount": number
        }
    ]
}
```

Pour obtenir des exemples d'autres AWS CLI commandes que vous pouvez utiliser pour vos tâches Gestionnaire de correctifs de configuration, consultez Utilisation dGestionnaire de correctifs' AWS CLI () (p. 1147) .

AWS Systems Manager Distributor

vous AWS Systems Manager Distributor permet d'empaqueter vos propres logiciels et de trouver les packages logiciels d'agent fournis par AWS AmazonCloudWatchAgent , tels que , ou des packages tiers tels que Trend Micro, à installer sur les instances gérées. AWS Systems Manager publie des ressources, telles que des packages logiciels, sur les instances Distributor gérées.AWS Systems Manager La publication d'un package publie des versions spécifiques du document du package (un Systems Manager document (p. 1211) que vous créez lorsque vous ajoutez le package dans Distributor) sur les instances gérées que vous identifiez par le IDs compte AWS , les balises ou une IDs région .AWS

Une fois que vous avez créé un package dans Distributor , qui crée un AWS Systems Manager document , vous pouvez installer le package de l'une des manières suivantes :

- 1 fois en utilisantAWS Systems Manager Fonctionnalité Exécuter la commande (p. 941)
- On a schedule by using (Selon un calendrier en utilisant)AWS Systems Manager State Manager (p. 987)

> **Important**
>
> Distributorne garantit pas que les packages tiers que vous installez sont exempts de tout programme malveillant potentiel. Nous vous encourageons à effectuer vos propres vérifications supplémentaires pour garantir la conformité à vos contrôles de sécurité internes. La sécurité est une responsabilité partagée entre AWS et vous-même. Ceci est décrit comme le modèle de responsabilité partagée. Pour en savoir plus, consultez le modèle de responsabilité partagée.

Comment mon organisation peut-elle tirer parti de Distributor ?

Distributor offre les avantages suivants :

- Un package, de nombreuses plateformes

  Des fichiers .zip peuvent être joints à un document qui est installé sur différents systèmes d'exploitation (par exemple, Windows, Ubuntu Server , ou Debian Server).Red Hat Enterprise Linux Pour de plus amples informations sur les plateformes prises en charge, veuillez consulter Architectures et plateformes de package prises en charge (p. 1176).
- Contrôle de l'accès au package à partir de groupes d'instances gérées

  Vous pouvez utiliser Fonctionnalité Exécuter la commande ou State Manager pour contrôler quelles sont vos instances gérées qui possèdent un package et la version de ce package. Les instances gérées peuvent être regroupées par instance, numéros de IDs compte, balises ou AWS régions.AWS Vous pouvez utiliser des associations State Manager pour diffuser différentes versions d'un package à différents groupes d'instances.
- De nombreux packages dAWS'agent inclus et prêts à l'emploi

  Distributor comprend de nombreux packages d'agent AWS qui sont prêts à être déployés sur des instances gérées. Recherchez des packages dans la page de liste Packages Distributor qui est publiée par Amazon. Exemples : AmazonCloudWatchAgent et AWSPVDriver.
- Automatiser le déploiement

  Pour maintenir à jour votre environnement, utilisez State Manager pour planifier le déploiement automatique des packages sur les instances cible lors du premier lancement de ces dernières.

À qui est destiné Distributor ?

- Tout client AWS qui souhaite créer de nouveaux packages logiciels ou déployer des packages logiciels existants, notamment des packages publiés par AWS, sur plusieurs instances gérées AWS Systems Manager à un moment donné.
- Les développeurs de logiciels qui créent des packages logiciels.
- Les administrateurs qui sont responsables de la mise à jour des instances gérées AWS Systems Manager avec les tous derniers packages logiciels.

Quelles sont les fonctions d'Distributor ?

- Déploiement de packages sur les instances Windows et Linux

  Distributor vous permet de déployer des packages logiciels sur des instances EC2 pour Linux et Windows Server. Pour obtenir une liste des types de systèmes d'exploitation d'instances pris en charge, consultez the section called “Architectures et plateformes de package prises en charge” (p. 1176).
- Déploiement de packages une seule fois ou selon un calendrier automatisé

  Vous pouvez choisir de déployer les packages une seule fois, selon une planification régulière, ou chaque fois que la version de package par défaut est remplacée par une autre version.
- Réinstallation complète des packages ou mises à jour sur place

  Pour installer une nouvelle version de package, vous pouvez désinstaller complètement la version actuelle et en installer une nouvelle à la place. Vous pouvez également seulement mettre à jour la version actuelle avec des composants nouveaux et mis à jour, conformément à un script de mise à jour que vous fournissez. Votre application de package n'est pas disponible lors d'une réinstallation, mais elle peut rester disponible lors d'une mise à jour sur place. Les mises à jour sur place sont particulièrement utiles pour les applications de surveillance de la sécurité ou d'autres scénarios dans lesquels vous devez éviter les temps d'arrêt des applications.

- Accès aux PowerShell fonctionnalités de la console, de l'interface de ligne de commande, de l' et du kit SDKDistributor

  Vous pouvez utiliser Distributor via la console AWS Systems Manager, l'AWS CLI, les Outils AWS pour PowerShell ou le kit SDK AWS de votre choix.
- IAM contrôle d'accès

  En utilisant les stratégies IAM, vous pouvez contrôler les membres de votre organisation qui sont autorisés à créer, mettre à jour, déployer ou supprimer des packages ou des versions de package. Par exemple, vous pouvez accorder à un administrateur l'autorisation de déployer des packages, mais pas celle de modifier les packages ou de créer de nouvelles versions de package.
- Prise en charge de la capacité de journalisation et d'audit

  Vous pouvez contrôler et consigner les actions des utilisateurs de Distributor dans votre compte AWS via l'intégration avec d'autres services AWS. Pour plus d'informations, consultez la section Audit et journalisation de l'activité de Distributor (p. 1208).

Qu'est-ce qu'un package ?

Un package est un ensemble de logiciels ou de ressources installables, qui inclut les éléments suivants :

- Un fichier .zip de logiciels par plateforme de système d'exploitation cible. Chaque fichier .zip doit contenir les éléments suivants :
  - Un install et un uninstall script . Les instances basées sur nécessitent Windows Server des scripts (scripts nommés PowerShell et `install.ps1`).`uninstall.ps1` Les instances basées sur Linux exigent des scripts shell (scripts nommés `install.sh` et `uninstall.sh`). L'Agent SSM lit et exécute les instructions des scripts install et uninstall.
  - Un fichier exécutable. L'Agent SSM doit trouver ce fichier exécutable pour installer le package sur les instances cible.
- Un fichier manifeste au format JSON qui décrit le contenu du package. Le manifeste n'est pas inclus dans le fichier .zip, mais il est stocké dans le même compartiment Amazon S3 que les fichiers .zip qui constituent le package. Le manifeste identifie la version du package et mappe les fichiers .zip du package aux attributs d'instance cible, tels que la version du système d'exploitation ou l'architecture. Pour de plus amples informations sur la création du manifeste, veuillez consulter Étape 2 : Créer le manifeste du package JSON (p. 1185).

Lorsque vous choisissez le package de création Simple dans la console Distributor, Distributor génère les scripts d'installation et de désinstallation, les hachages de fichier, ainsi que le manifeste de package JSON pour vous, en fonction du nom du fichier exécutable de logiciel, et des plateformes et architectures cibles.

Architectures et plateformes de package prises en charge

Distributor prend en charge la distribution du package à n'importe quelle version des plateformes suivantes, du moment qu'elle est prise en charge en tant qu'instance gérée Systems Manager. Une valeur de version doit correspondre à la version exacte de l'AMI du système d'exploitation que vous ciblez. Pour de plus amples informations sur la détermination de cette version, veuillez consulter l'étape 4 de Étape 2 : Créer le manifeste du package JSON (p. 1185).

Plateforme	Valeur de code dans le fichier manifeste	Architecture
Windows Server	`windows`	`x86_64ou386`
Debian Server	`debian`	`x86_64ou386`

Plateforme	Valeur de code dans le fichier manifeste	Architecture
Ubuntu Server	`ubuntu`	`x86_64`ou`386` `arm64` (Ubuntu Server 16 et versions ultérieures, types d'instances A1)
Red Hat Enterprise Linux (RHEL)	`redhat`	`x86_64`ou`386` `arm64` (RHEL 7.6 et versions ultérieures, types d'instances A1)
CentOS	`centos`	`x86_64`ou`386`
Amazon Linux et Amazon Linux 2	`amazon`	`x86_64`ou`386` `arm64` (Amazon Linux 2, types d'instance A1)
SUSE Linux Enterprise Server (SLES)	`suse`	`x86_64`ou`386`
openSUSE	`opensuse`	`x86_64`ou`386`
openSUSE	`opensuseleap`	`x86_64`ou`386`
Oracle Linux	`oracle`	`x86_64`

Rubriques

- Configuration d'Distributor (p. 1177)
- Utilisation des Distributor (p. 1179)
- Audit et journalisation de l'activité de Distributor (p. 1208)
- Résolution des problèmes liés à AWS Systems Manager Distributor (p. 1209)

Configuration d'Distributor

Avant d'utiliser Distributor pour créer, gérer et déployer des packages logiciels, exécutez les étapes ci-après.

Rubriques

- Étape 1 : Exécuter les opérations Distributor prérequises (p. 1177)
- Étape 2 : Vérifier ou créer un profil d'instance IAM avec des autorisations Distributor (p. 1178)
- Étape 3 : Contrôle de l'accès utilisateur aux packages (p. 1179)
- Étape 4 : Créer ou choisir un Amazon S3 compartiment (p. 1179)

Étape 1 : Exécuter les opérations Distributor prérequises

Avant d'utiliser Distributor, vérifiez que votre environnement respecte les conditions requises suivantes.

Prérequis d'Distributor

Exigence	Description
Agent SSM	Agent SSM version 2.3.274.0 ou ultérieure doit être installé sur les instances sur lesquelles vous souhaitez déployer ou supprimer des packages. Pour installer ou mettre à jour Agent SSM, consultez Utilisation des Agent SSM (p. 72).
AWS CLI	(Facultatif) Pour utiliser l'AWS CLI à la place de la console AWS Systems Manager pour créer et gérer des packages, installez la dernière version de l'AWS CLI sur votre ordinateur local. Pour de plus amples informations sur l'installation ou la mise à niveau de l'interface de ligne de commande, veuillez consulter Installation de l'AWS Command Line Interface dans le AWS Command Line Interface Guide de l'utilisateur.
Outils AWS pour PowerShell	(Facultatif) Pour utiliser les Outils pour PowerShell à la place de la console AWS Systems Manager pour créer et gérer des packages, installez la dernière version des Outils pour PowerShell sur votre ordinateur local. Pour plus d'informations sur l'installation ou la mise à niveau de l' Outils pour PowerShell , consultez Configuration des outils AWS pour Windows PowerShell ou des outils AWS pour PowerShell Core dans le Guide Outils AWS pour PowerShell de l'utilisateur .

Note

ne prend Systems Manager actuellement pas en charge la distribution des packages aux Oracle Linux instances à l'aide d' Distributor .

Étape 2 : Vérifier ou créer un profil d'instance IAM avec des autorisations Distributor

Par défaut, AWS Systems Manager n'a pas le droit d'effectuer des actions sur vos instances. Vous devez accorder l'accès à l'aide d'un profil d'instance IAM. Un profil d'instance est un conteneur qui transmet des informations sur le rôle IAM à une instance EC2 lors du lancement. Cette exigence s'applique aux autorisations relatives à toutes les fonctionnalités AWS Systems Manager, et non pas uniquement aux fonctionnalités de Distributor.

Si vous utilisez déjà d'autres fonctionnalités Systems Manager, comme par exemple Fonctionnalité Exécuter la commande et State Manager, un profil d'instance avec les autorisations requises pour Distributor est déjà attaché à vos instances. La façon la plus simple de vous assurer que vous disposez des autorisations nécessaires pour effectuer Distributor des tâches est d'attacher la AmazonSSMManagedInstanceCore stratégie à votre profil d'instance. Pour plus d'informations, consultez la section Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34).

Étape 3 : Contrôle de l'accès utilisateur aux packages

Les stratégies IAM vous permettent de contrôler les personnes pouvant créer, déployer et gérer des packages. Vous pouvez également contrôler les actions d'API Fonctionnalité Exécuter la commande et State Manager que les utilisateurs peuvent exécuter sur les instances gérées.

Format ARN

Les packages définis par l'utilisateur sont associés à un document ARNs et ont le format suivant :

```
arn:aws:ssm:region-id:account-id:document/document-name
```

Voici un exemple pour .

```
arn:aws:ssm:us-west-1:123456789012:document/ExampleDocumentName
```

Vous pouvez utiliser une paire de stratégies IAM par défaut fournies par AWS (une pour les utilisateurs finaux et une pour les administrateurs) afin d'accorder les autorisations relatives aux activités Distributor. Vous pouvez également créer des stratégies IAM personnalisées appropriées, répondant à vos exigences en matière d'autorisations.

Pour plus d'informations sur l'utilisation des variables dans les stratégies IAM, consultez Éléments IAM de stratégie : Variables.

Pour plus d'informations sur la création de stratégies et leur liaison avec des utilisateurs ou des groupes IAM, consultez Création de stratégies IAM et Ajout et suppression de stratégies IAM dans le IAM Guide de l'utilisateur.

Étape 4 : Créer ou choisir un Amazon S3 compartiment

Lorsque vous créez un package en utilisant le flux de travail Simple dans la console, vous choisissez un compartiment S3 existant dans lequel Distributor charge votre logiciel. Dans le flux de travail Advanced (Avancé), vous devez charger les fichiers .zip de vos logiciels ou ressources dans un compartiment S3 avant de commencer. Si vous créez un package à l'aide des flux de travail Simple ou Advanced (Avancé) dans la console, ou à l'aide de l'API, vous devez avoir un compartiment S3 avant de commencer à créer votre package. Dans le cadre du processus de création de package, Distributor copie vos logiciels et ressources à installer à partir de ce compartiment vers un magasin AWS Systems Manager interne. Étant donné que les ressources sont copiées vers un magasin interne, vous pouvez supprimer ou réutiliser votre compartiment S3 lors de la création du package est terminée.

Pour plus d'informations sur la création d'un compartiment, consultez Créer un compartiment dans le Guide de mise en route Amazon Simple Storage Service. Pour de plus amples informations sur l'exécution d'une commande de l'AWS CLI pour créer un compartiment, veuillez consulter mb dans la Référence des commandes de l'AWS CLI.

Utilisation des Distributor

Vous pouvez utiliser la AWS Systems Manager console , les outils de ligne de commande AWS (AWS CLI et Outils AWS pour PowerShell) et AWS SDKs pour ajouter, gérer ou déployer des packages dans Distributor . Avant d'ajouter un package à Distributor :

- Créez et compressez les ressources à installer.
- (Facultatif) Créez un fichier manifeste JSON pour le package. Il n'est pas obligatoire d'utiliser le processus de création de package Simple dans la console Distributor. La création du package Simple génère un fichier manifeste JSON pour vous.

  Vous pouvez utiliser la console AWS Systems Manager, un éditeur de texte ou un éditeur JSON pour créer le fichier manifeste.

- Vous devez avoir un compartiment S3 prêt pour stocker vos ressources ou logiciels installables. Si vous utilisez le processus de création de package Advanced (Avancé), chargez vos ressources dans le compartiment S3 avant de commencer.

  Note

  Vous pouvez supprimer ou réutiliser ce compartiment après avoir créé votre package, car Distributor déplace le contenu du package vers un compartiment Systems Manager interne dans le cadre du processus de création de package.

Les packages publiés par AWS sont déjà conditionnés et prêts pour le déploiement. Pour déployer un package publié par AWS sur des instances gérées, consultez Installer ou mettre à jour des packages (p. 1198).

Rubriques

- Afficher les packages (p. 1180)
- Créer un package (p. 1182)
- Modifier les autorisations du package (console) (p. 1193)
- Modifier les balises du package (console) (p. 1193)
- Ajouter une version de package à Distributor (p. 1194)
- Installer ou mettre à jour des packages (p. 1198)
- Désinstaller un package (p. 1204)
- Supprimer un package (p. 1205)

Afficher les packages

Pour afficher les packages qui sont disponibles pour l'installation, vous pouvez utiliser la AWS Systems Manager console ou votre outil de ligne de commande AWS préféré. Pour accéder à Distributor , ouvrez la AWS Systems Manager console et choisissez Distributor dans le volet de navigation de gauche. Vous verrez tous les packages disponibles.

La section suivante décrit comment afficher les Distributor packages à l'aide de votre outil de ligne de commande préféré.

Affichage des packages (ligne de commande)

Cette section contient des informations sur la façon dont vous pouvez utiliser votre outil de ligne de commande préféré pour afficher les Distributor packages à l'aide des commandes fournies.

Linux

Pour afficher les packages à l'aide de l' AWS CLI sous Linux

- Pour afficher tous les packages, à l'exception des packages partagés, exécutez la commande suivante.

```
aws ssm list-documents \
    --filters Key=DocumentType,Values=Package
```

- Pour afficher tous les packages détenus par Amazon, exécutez la commande suivante.

```
aws ssm list-documents \
    --filters Key=DocumentType,Values=Package Key=Owner,Values=Amazon
```

- Pour afficher tous les packages détenus par des tiers, exécutez la commande suivante.

```
aws ssm list-documents \
    --filters Key=DocumentType,Values=Package Key=Owner,Values=ThirdParty
```

Windows

Pour afficher les packages à l'aide de l' AWS CLI sous Windows

- Pour afficher tous les packages, à l'exception des packages partagés, exécutez la commande suivante.

```
aws ssm list-documents ^
    --filters Key=DocumentType,Values=Package
```

- Pour afficher tous les packages détenus par Amazon, exécutez la commande suivante.

```
aws ssm list-documents ^
    --filters Key=DocumentType,Values=Package Key=Owner,Values=Amazon
```

- Pour afficher tous les packages détenus par des tiers, exécutez la commande suivante.

```
aws ssm list-documents ^
    --filters Key=DocumentType,Values=Package Key=Owner,Values=ThirdParty
```

PowerShell

Pour afficher les packages à l'aide de l'Outils pour PowerShell

- Pour afficher tous les packages, à l'exception des packages partagés, exécutez la commande suivante.

```
$filter = New-Object Amazon.SimpleSystemsManagement.Model.DocumentKeyValuesFilter
$filter.Key = "DocumentType"
$filter.Values = "Package"

Get-SSMDocumentList `
    -Filters @($filter)
```

- Pour afficher tous les packages détenus par Amazon, exécutez la commande suivante.

```
$typeFilter = New-Object Amazon.SimpleSystemsManagement.Model.DocumentKeyValuesFilter
$typeFilter.Key = "DocumentType"
$typeFilter.Values = "Package"

$ownerFilter = New-Object
 Amazon.SimpleSystemsManagement.Model.DocumentKeyValuesFilter
$ownerFilter.Key = "Owner"
$ownerFilter.Values = "Amazon"

Get-SSMDocumentList `
    -Filters @($typeFilter,$ownerFilter)
```

- Pour afficher tous les packages détenus par des tiers, exécutez la commande suivante.

```
$typeFilter = New-Object Amazon.SimpleSystemsManagement.Model.DocumentKeyValuesFilter
$typeFilter.Key = "DocumentType"
$typeFilter.Values = "Package"
```

```
$ownerFilter = New-Object
 Amazon.SimpleSystemsManagement.Model.DocumentKeyValuesFilter
$ownerFilter.Key = "Owner"
$ownerFilter.Values = "ThirdParty"

Get-SSMDocumentList `
    -Filters @($typeFilter,$ownerFilter)
```

Créer un package

Pour créer un package, préparez vos logiciels ou ressources installables, un fichier ZIP par plateforme de système d'exploitation. Vous devez disposer d'au moins un fichier pour créer un package.

Différentes plateformes peuvent parfois utiliser le même fichier, mais tous les fichiers que vous attachez à votre package doivent être répertoriés dans la section `Files` du manifeste. Si vous créez un package en utilisant le flux de travail simple dans la console, le manifeste est généré pour vous. Le nombre maximum de fichiers que vous pouvez attacher à un seul document est de 20. La taille maximale d'un fichier est de 1 Go. Pour de plus amples informations sur les plateformes prises en charge, veuillez consulter Architectures et plateformes de package prises en charge (p. 1176).

Lorsque vous créez un package, vous ajoutez un nouveau document SSM (p. 1211). Ce document vous permet de déployer le package sur des instances gérées.

À des fins de démonstration uniquement, un exemple de package, ExamplePackage .zip, est disponible en téléchargement à partir de notre site web. L'exemple de package inclut un manifeste JSON complet et trois fichiers .zip contenant des programmes d'installation pour PowerShell v7.0.0. Les scripts d'installation et de désinstallation ne contiennent pas de commandes valides. Même si vous devez compresser chacun des fichiers installables de logiciel et des scripts dans un fichier .zip pour créer un package dans le flux de travail Advanced (Avancé), vous ne compressez pas les ressources installables dans le flux de travail Simple.

Rubriques

- Créer un package (simple) (p. 1182)
- Créer un package (avancé) (p. 1184)

Créer un package (simple)

Cette section décrit comment créer un package dans Distributor en choisissant le flux de travail de création de package Simple dans la console Distributor. Pour créer un package, préparez vos ressources installables, un fichier par plateforme de système d'exploitation. Vous devez disposer d'au moins un fichier pour créer un package. Le processus de création de package Simple génère des scripts d'installation et de désinstallation, des hachages de fichier et un manifeste au format JSON pour vous. Le flux de travail Simple gère le processus de chargement et de compression de vos fichiers installables, ainsi que la création d'un nouveau package et du document SSM (p. 1211) associé. Pour de plus amples informations sur les plateformes prises en charge, veuillez consulter Architectures et plateformes de package prises en charge (p. 1176).

Lorsque vous utilisez la méthode Simple pour créer un package, Distributor crée des scripts `install` et `uninstall`. Toutefois, lorsque vous créez un package pour une mise à jour sur place, vous devez fournir votre propre contenu de script `update` dans l'onglet Update script (Script de mise à jour). Lorsque vous ajoutez des commandes d'entrée pour un script `update`, Distributor inclut ce script dans le package .zip qu'il crée, avec les scripts `uninstall` et `install`.

> **Note**
>
> L'option de mise à jour `In-place` permet d'ajouter des fichiers nouveaux ou mis à jour à une installation de package existante sans mettre l'application associée hors connexion.

Pour créer un package (simple)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Distributor.
3. Sur la page d'accueil de Distributor, choisissez Create package (Créer un package), puis Simple.
4. Sur la page Create package (Créer un package), entrez un nom pour votre package. Les noms de package peuvent contenir des lettres, des chiffres, des points, des tirets et des traits de soulignement. Le nom doit être suffisamment générique pour pouvoir s'appliquer à toutes les versions des pièces jointes du package, mais suffisamment spécifique pour identifier l'objectif du package.
5. (Facultatif) Pour Version name (Nom de version), entrez un nom de version. Les noms de version peuvent comporter 512 caractères au maximum et ne peuvent pas contenir de caractères spéciaux.
6. Pour Location (Emplacement), choisissez un compartiment en utilisant le nom et le préfixe du compartiment ou en utilisant son URL.
7. Dans Upload software (Télécharger un logiciel), choisissez Add software (Ajouter un logiciel), puis recherchez les fichiers logiciels installables portant une extensions `.rpm`, `.msi` ou `.deb`. Vous pouvez charger plusieurs fichiers de logiciel en une seule action.
8. Pour Target platform (Plateforme cible), vérifiez que la plateforme de système d'exploitation cible pour chaque fichier installable est correcte. Si le système d'exploitation indiqué n'est pas correct, choisissez le système d'exploitation approprié dans la liste déroulante.

   Dans le flux de travail de création de package Simple, comme vous chargez chaque fichier installable une seule fois, des étapes supplémentaires sont nécessaires pour demander à Distributor de cibler un seul fichier sur plusieurs systèmes d'exploitation. Par exemple, si vous chargez un fichier de logiciel installable nommé `Logtool_v1.1.1.rpm`, vous devez modifier certaines valeurs par défaut dans le flux de travail Simple pour cibler le même logiciel sur les systèmes d'exploitation Amazon Linux et Ubuntu. Lorsque vous ciblez plusieurs plateformes, effectuez l'une des opérations suivantes.

   - Utilisez plutôt le flux de travail Advanced (Avancé), compressez chaque fichier installable en un fichier .zip avant de commencer et créez manuellement le manifeste afin qu'un fichier installable puisse cibler plusieurs plateformes ou versions de système d'exploitation. Pour plus d'informations, consultez la section Créer un package (avancé) (p. 1184).
   - Modifiez manuellement le fichier manifeste dans le flux de travail Simple pour que votre fichier .zip cible plusieurs plateformes ou versions de système d'exploitation. Pour plus d'informations sur la façon de procéder, consultez la fin de l'étape 4 dans Étape 2 : Créer le manifeste du package JSON (p. 1185).
9. Pour la version de plateforme, vérifiez que la version de plateforme de système d'exploitation affichée est **`_any`** , une version majeure suivie d'un caractère générique (7.*) ou la version exacte du système d'exploitation à laquelle vous souhaitez que votre logiciel s'applique. Pour plus d'informations sur la spécification d'une version de plateforme de système d'exploitation, consultez l'étape 4 de Étape 2 : Créer le manifeste du package JSON (p. 1185).
10. Pour Architecture, choisissez l'architecture de processeur correcte pour chaque fichier installable dans la liste déroulante. Pour plus d'informations sur les architectures de processeur prises en charge, consultez Architectures et plateformes de package prises en charge (p. 1176).
11. (Facultatif) Développez Scripts et vérifiez les scripts générés par Distributor pour votre logiciel installable.
12. (Facultatif) Pour fournir un script de mise à jour à utiliser avec les mises à jour sur place, développez Scripts, choisissez l'onglet Update script (Script de mise à jour) et entrez vos commandes de script de mise à jour.

    Systems Manager ne génère pas de scripts de mise à jour en votre nom.
13. Pour ajouter d'autres fichiers de logiciels installables, choisissez Add software (Ajouter des logiciels). Sinon, accédez à l'étape suivante.

14. (Facultatif) Développez Manifest (Manifeste) et vérifiez le manifeste de package JSON généré par Distributor pour vos logiciels installables. Si vous avez modifié des informations relatives à vos logiciels depuis que vous avez commencé cette procédure, comme la version de plateforme ou la plateforme cible, choisissez Generate manifest (Générer un manifeste) pour afficher le manifeste de package mis à jour.

    Vous pouvez modifier le manifeste manuellement si vous souhaitez cibler un logiciel installable pour plusieurs systèmes d'exploitation, comme décrit à l'étape 8. Pour plus d'informations sur la modification du manifeste, consultez Étape 2 : Créer le manifeste du package JSON (p. 1185).
15. Choisissez Create package (Créer un package).

Attendez que Distributor finisse de charger vos logiciels et de créer votre package. Distributor indique le statut de chargement pour chaque fichier installable. Selon le nombre et la taille des packages que vous ajoutez, cela peut prendre quelques minutes. Distributor vous redirige vers la page Package details (Détails du package) pour le nouveau package, mais vous pouvez choisir d'ouvrir cette page vous-même une fois les logiciels chargés. La page Package details (Détails du package) n'affiche pas toutes les informations sur votre package tant que Distributor n'a pas terminé le processus de création de ce package. Pour arrêter le processus de chargement et de création de package, choisissez Cancel (Annuler).

Si Distributor ne peut pas charger les fichiers de logiciels installables, il affiche un message Upload failed (Échec du chargement). Pour relancer le chargement, choisissez Retry upload (Réessayer le chargement). Pour plus d'informations sur la façon de résoudre les échecs de création de package, consultez Résolution des problèmes liés à AWS Systems Manager Distributor (p. 1209).

Créer un package (avancé)

Dans cette section, découvrez comment les utilisateurs avancés peuvent créer un package dans Distributor après avoir chargé des ressources installables compressées avec des scripts d'installation et de désinstallation, ainsi qu'un fichier manifeste JSON dans un compartiment S3.

Pour créer un package, préparez vos fichiers .zip de ressources installables, un fichier .zip par plateforme de système d'exploitation. Vous devez disposer d'au moins un fichier .zip pour créer un package. Créez ensuite un manifeste JSON. Le manifeste inclut des pointeurs vers vos fichiers de code de package. Une fois que les fichiers de code requis ont été ajoutés à un dossier et que le manifeste a été renseigné avec des valeurs correctes, chargez votre package dans un compartiment S3.

Un exemple de package, ExamplePackage .zip, est disponible en téléchargement à partir de notre site web. Cet exemple de package comprend un manifeste JSON complet et trois fichiers .zip.

Rubriques

- Étape 1 : Créer les fichiers ZIP (p. 1184)
- Étape 2 : Créer le manifeste du package JSON (p. 1185)
- Étape 3 : Charger le package et le manifeste dans un compartiment S3 (p. 1190)
- Étape 4 : Ajouter un package à Distributor (p. 1191)

Étape 1 : Créer les fichiers ZIP

Votre package doit reposer sur au moins un fichier .zip de logiciels ou de ressources installables. Un package inclut un fichier .zip par système d'exploitation que vous souhaitez prendre en charge, sauf si un fichier .zip peut être installé sur plusieurs systèmes d'exploitation. Par exemple, les instances Red Hat Enterprise Linux et Amazon Linux peuvent généralement exécuter les mêmes fichiers exécutables .RPM. Par conséquent, vous n'avez besoin d'attacher qu'un seul fichier .zip à votre package pour pouvoir prendre en charge les deux systèmes d'exploitation.

Fichiers requis

Les éléments suivants doivent obligatoirement être présents dans chaque fichier .zip :

- Un install et un uninstall script . Les instances basées sur nécessitent Windows Server des scripts (scripts nommés PowerShell et `install.ps1`).`uninstall.ps1` Les instances basées sur Linux exigent des scripts shell (scripts nommés `install.sh` et `uninstall.sh`). L'Agent SSM exécute les instructions des scripts install et uninstall.

  Par exemple, vos scripts d'installation peuvent exécuter un programme d'installation (par exemple, .rpm ou .msi), ils peuvent copier des fichiers ou définir des paramètres de configuration.
- Un fichier exécutable, des packages de programme d'installation (.rpm, .deb, .msi, etc.), d'autres scripts ou fichiers de configuration.

Fichiers facultatifs

L'élément suivant est facultatif dans chaque fichier .zip :

- Un script update. Le fait de fournir un script de mise à jour permet d'utiliser l'option `In-place update` pour installer un package. Lorsque vous souhaitez ajouter des fichiers nouveaux ou mis à jour à une installation de package existante, l'`In-place update`option ne met pas l'application de package hors ligne pendant la mise à jour. Les instances basées sur nécessitent un Windows Server script (script nommé PowerShell).`update.ps1` Les instances basées sur Linux nécessitent un script shell (script nommé `update.sh`). Agent SSM exécute les instructions du script update.

Pour de plus amples informations sur l'installation ou la mise à jour de packages, veuillez consulter Installer ou mettre à jour des packages (p. 1198).

Pour obtenir des exemples de fichiers .zip, y compris install des exemples uninstall et des scripts, téléchargez l'exemple de package, ExamplePackage .zip.

Étape 2 : Créer le manifeste du package JSON

Une fois que vous avez préparé et compressé vos fichiers installables, créez un manifeste JSON. Vous trouverez ci-après un modèle. Les parties du modèle de manifeste sont décrites dans la procédure présentée dans cette section. Vous pouvez utiliser un éditeur JSON pour créer ce manifeste dans un fichier distinct. Vous pouvez aussi créer le manifeste dans la console AWS Systems Manager lorsque vous créez un package.

```
{
  "schemaVersion": "2.0",
  "version": "your-version",
  "publisher": "optional-publisher-name",
  "packages": {
    "platform": {
      "platform-version": {
        "architecture": {
          "file": ".zip-file-name-1.zip"
        }
      }
    },
    "another-platform": {
      "platform-version": {
        "architecture": {
          "file": ".zip-file-name-2.zip"
        }
      }
    },
    "another-platform": {
      "platform-version": {
        "architecture": {
          "file": ".zip-file-name-3.zip"
```

```
        }
      }
    }
  },
  "files": {
    ".zip-file-name-1.zip": {
      "checksums": {
        "sha256": "checksum"
      }
    },
    ".zip-file-name-2.zip": {
      "checksums": {
        "sha256": "checksum"
      }
    }
  }
}
```

Pour créer un manifeste de package JSON

1. Ajoutez la version du schéma à votre manifeste. Dans cette édition, la version du schéma est toujours `2.0`.

   ```
   { "schemaVersion": "2.0",
   ```

2. Ajoutez une version de package définie par l'utilisateur à votre fichier manifeste. Il s'agit également de la valeur de Version name (Nom de la version) que vous spécifiez lorsque vous ajoutez votre package à Distributor. Cette version fait partie du document AWS Systems Manager créé par Distributor lorsque vous ajoutez votre package. Vous pouvez également fournir cette valeur sous forme d'entrée dans le document `AWS-ConfigureAWSPackage` pour installer une version du package autre que la plus récente. Une valeur `version` peut contenir des lettres, des chiffres, des traits de soulignement, des tirets et des points, et peut comporter un maximum de 128 caractères. Nous vous recommandons d'utiliser une version de package lisible par l'utilisateur afin qu'il soit plus facile pour vous et pour les autres administrateurs de spécifier les versions de packages exactes lorsque vous effectuez le déploiement. Voici un exemple pour .

   ```
   "version": "1.0.1",
   ```

3. (Facultatif) Ajoutez un nom d'éditeur. Voici un exemple pour .

   ```
   "publisher": "MyOrganization",
   ```

4. Ajoutez des packages. La section `"packages"` décrit les plateformes, versions et architectures prises en charge par les fichiers .zip dans votre package. Pour plus d'informations, consultez la section Architectures et plateformes de package prises en charge (p. 1176).

   L' *`platform-version`* peut être la valeur générique, `_any` . Utilisez-le pour indiquer qu'un fichier .zip prend en charge n'importe quelle version de la plateforme. Vous pouvez également spécifier une version majeure suivie d'un caractère générique afin que toutes les versions mineures soient prises en charge, par exemple 7.*. Si vous choisissez de spécifier un *`platform-version`* pour une version spécifique du système d'exploitation, assurez-vous qu'elle correspond à la version exacte de l'AMI du système d'exploitation que vous ciblez. Voici les ressources suggérées pour obtenir la valeur correcte du système d'exploitation.

   - Sur une instance basée sur Windows Server, la version est disponible en tant que données Windows Management Instrumentation (WMI). Vous pouvez exécuter la commande d'invite de commande suivante sur une instance basée sur , pour obtenir les informations de version, puis analyser les résultats obtenus pour Windows Server `.version` Cette commande n'affiche pas la version pour Windows Server Nano ; la valeur de version pour Windows Server Nano est `nano` .

```
wmic OS get /format:list
```

- Sur une instance Linux, vous pouvez obtenir la version en commençant par rechercher la version du système d'exploitation (la commande suivante). Recherchez la valeur de `VERSION_ID`.

  ```
  cat /etc/os-release
  ```

  Si vous n'obtenez pas les résultats dont vous avez besoin, exécutez la commande suivante pour obtenir des informations de version LSB à partir du fichier `/etc/lsb-release` et recherchez la valeur de `DISTRIB_RELEASE`.

  ```
  lsb_release -a
  ```

  Si ces méthodes échouent, vous pouvez généralement trouver la version en fonction de la distribution. Par exemple, sous Debian Server , vous pouvez analyser le `/etc/debian_version` fichier, ou sur Red Hat Enterprise Linux , le `/etc/redhat-release` fichier.

  ```
  hostnamectl
  ```

```
"packages": {
    "platform": {
      "platform-version": {
        "architecture": {
          "file": ".zip-file-name-1.zip"
        }
      }
    },
    "another-platform": {
      "platform-version": {
        "architecture": {
          "file": ".zip-file-name-2.zip"
        }
      }
    },
    "another-platform": {
      "platform-version": {
        "architecture": {
          "file": ".zip-file-name-3.zip"
        }
      }
    }
  }
```

Voici un exemple pour . Dans cet exemple, la plateforme du système d'exploitation est `amazon`, la version prise en charge est `2016.09`, l'architecture est `x86_64` et le fichier .zip qui prend en charge cette plateforme est `test.zip`.

```
{
    "amazon": {
        "2016.09": {
            "x86_64": {
                "file": "test.zip"
            }
        }
    }
```

```
},
```

Vous pouvez ajouter la valeur générique `_any` pour indiquer que le package prend en charge toutes les versions de l'élément parent. Par exemple, pour indiquer que le package est pris en charge sur n'importe quelle version d'Amazon Linux, votre déclaration de package doit être similaire à ce qui suit. Vous pouvez utiliser le caractère générique `_any` aux niveaux de la version ou de l'architecture pour prendre en charge toutes les versions d'une plateforme ou toutes les architectures dans une version, ou encore toutes les versions et architectures d'une plateforme.

```
{
    "amazon": {
        "_any": {
            "x86_64": {
                "file": "test.zip"
            }
        }
    }
},
```

L'exemple suivant ajoute `_any` pour montrer que le premier package, `data1.zip`, est pris en charge pour toutes les architectures d'Amazon Linux 2016.09. Le deuxième package, `data2.zip`, est pris en charge pour toutes les versions d'Amazon Linux, mais uniquement pour les instances dotées de l'architecture `x86_64`. Les `2016.09` versions `_any` et sont des entrées sous `amazon` . Il existe une plateforme (Amazon Linux), mais différentes versions et architectures prises en charge, ainsi que les fichiers .zip associés.

```
{
    "amazon": {
        "2016.09": {
            "_any": {
                "file": "data1.zip"
            }
        },
        "_any": {
            "x86_64": {
                "file": "data2.zip"
            }
        }
    }
}
```

Vous pouvez faire référence à un fichier .zip plusieurs fois dans la section `"packages"` du manifeste, si le fichier .zip prend en charge plusieurs plateformes. Par exemple, si vous avez un fichier .zip qui prend en charge à la fois les versions Red Hat Enterprise Linux 7.x et Amazon Linux , vous avez deux entrées dans la `"packages"` section qui pointent vers le même fichier .zip, comme illustré dans l'exemple suivant.

```
{
    "amazon": {
        "2018.03": {
            "x86_64": {
                "file": "test.zip"
            }
        }
    },
    "redhat": {
        "7.*": {
            "x86_64": {
                "file": "test.zip"
            }
```

```
        }
    }
},
```

5. Ajoutez la liste des fichiers .zip qui font partie de ce package à partir de l'étape 4. Chaque entrée de fichier requiert le nom de fichier et le total de contrôle de la valeur de hachage `sha256`. Les valeurs de total de contrôle dans le manifeste doivent correspondre à la valeur de hachage `sha256` dans les ressources compressées pour empêcher l'échec de l'installation des packages.

   Pour obtenir le total de contrôle exact de vos ressources installables, vous pouvez exécuter les commandes suivantes. Sous Linux, exécutez `cat file-name.zip | openssl dgst -sha256` . Sous Windows, exécutez l'`Get-FileHash -Path path-to-.zip-file`applet de commande dans PowerShell .

   La section `"files"` du manifeste inclut une référence à chacun des fichiers .zip de votre package.

```
"files": {
        "test-agent-x86.deb.zip": {
            "checksums": {
                "sha256":
 "EXAMPLE2706223c7616ca9fb28863a233b38e5a23a8c326bb4ae241dcEXAMPLE"
            }
        },
        "test-agent-x86_64.deb.zip": {
            "checksums": {
                "sha256":
 "EXAMPLE572a745844618c491045f25ee6aae8a66307ea9bff0e9d1052EXAMPLE"
            }
        },
        "test-agent-x86_64.nano.zip": {
            "checksums": {
                "sha256":
 "EXAMPLE63ccb86e830b63dfef46995af6b32b3c52ce72241b5e80c995EXAMPLE"
            }
        },
        "test-agent-rhel5-x86.nano.zip": {
            "checksums": {
                "sha256":
 "EXAMPLE13df60aa3219bf117638167e5bae0a55467e947a363fff0a51EXAMPLE"
            }
        },
        "test-agent-x86.msi.zip": {
            "checksums": {
                "sha256":
 "EXAMPLE12a4abb10315aa6b8a7384cc9b5ca8ad8e9ced8ef1bf0e5478EXAMPLE"
            }
        },
        "test-agent-x86_64.msi.zip": {
            "checksums": {
                "sha256":
 "EXAMPLE63ccb86e830b63dfef46995af6b32b3c52ce72241b5e80c995EXAMPLE"
            }
        },
        "test-agent-rhel5-x86.rpm.zip": {
            "checksums": {
                "sha256":
 "EXAMPLE13df60aa3219bf117638167e5bae0a55467e947a363fff0a51EXAMPLE"
            }
        },
        "test-agent-rhel5-x86_64.rpm.zip": {
            "checksums": {
                "sha256":
 "EXAMPLE7ce8a2c471a23b5c90761a180fd157ec0469e12ed38a7094d1EXAMPLE"
            }
```

```
            }
        }
```

6. Une fois que vous avez ajouté vos informations de package, enregistrez et fermez le fichier manifeste.

Voici un exemple de fichier manifeste terminé. Dans cet exemple, vous avez un fichier .zip, `NewPackage_LINUX.zip`, qui prend en charge plusieurs plateformes, mais qui n'est référencé qu'une seule fois dans la section `"files"`.

```
{
    "schemaVersion": "2.0",
    "version": "1.7.1",
    "publisher": "Amazon Web Services",
    "packages": {
        "windows": {
            "_any": {
                "x86_64": {
                    "file": "NewPackage_WINDOWS.zip"
                }
            }
        },
        "amazon": {
            "_any": {
                "x86_64": {
                    "file": "NewPackage_LINUX.zip"
                }
            }
        },
        "ubuntu": {
            "_any": {
                "x86_64": {
                    "file": "NewPackage_LINUX.zip"
                }
            }
        }
    },
    "files": {
        "NewPackage_WINDOWS.zip": {
            "checksums": {
                "sha256":
 "EXAMPLEc2c706013cf8c68163459678f7f6daa9489cd3f91d52799331EXAMPLE"
            }
        },
        "NewPackage_LINUX.zip": {
            "checksums": {
                "sha256":
 "EXAMPLE2b8b9ed71e86f39f5946e837df0d38aacdd38955b4b18ffa6fEXAMPLE"
            }
        }
    }
}
```

Exemple de package

Un exemple de package, ExamplePackage .zip, est disponible en téléchargement à partir de notre site web. Cet exemple de package comprend un manifeste JSON complet et trois fichiers .zip.

Étape 3 : Charger le package et le manifeste dans un compartiment S3

Préparez votre package en copiant ou en déplaçant tous les fichiers .zip dans un dossier ou un répertoire. Un package valide nécessite le manifeste que vous avez créé à l'Étape 2 : Créer le manifeste du package JSON (p. 1185) et tous les fichiers .zip identifiés dans la liste de fichiers du manifeste.

Pour charger le package et le manifeste dans Amazon S3

1. Copiez ou déplacez dans un dossier ou un répertoire tous les fichiers d'archive .zip que vous avez spécifiés dans le manifeste. Ne compressez pas le dossier ou le répertoire dans lequel vous déplacez vos fichiers d'archive .zip et votre fichier manifeste.
2. Créez un compartiment ou choisissez un compartiment existant. Pour de plus amples informations, veuillez consulter Créer un compartiment dans le Guide de démarrage d'Amazon Simple Storage Service. Pour de plus amples informations sur l'exécution d'une commande de l'AWS CLI pour créer un compartiment, veuillez consulter mb dans la Référence des commandes de l'AWS CLI.
3. Chargez le dossier dans le compartiment. Pour de plus amples informations, veuillez consulter Ajouter un objet à un compartiment dans le Guide de démarrage d'Amazon Simple Storage Service. Si vous prévoyez de coller votre manifeste JSON dans la console AWS Systems Manager, ne chargez pas le fichier manifeste. Pour de plus amples informations sur l'exécution d'une commande de l'AWS CLI pour charger des fichiers dans un compartiment, veuillez consulter mv dans la Référence des commandes de l'AWS CLI.
4. Sur la page d'accueil du compartiment, choisissez le dossier que vous avez chargé. Si vous avez chargé vos fichiers dans un sous-dossier au sein d'un compartiment, veillez à noter le sous-dossier (également connu sous le nom de préfixe). Vous avez besoin du préfixe pour ajouter votre package dans Distributor.

Étape 4 : Ajouter un package à Distributor

Vous pouvez utiliser la AWS Systems Manager console , les outils de ligne de commande AWS (AWS CLI et Outils AWS pour PowerShell) ou AWS SDKs pour ajouter un nouveau package dans AWS Systems Manager Distributor . Lorsque vous ajoutez un package, vous ajoutez un nouveau document SSM (p. 1211). Ce document vous permet de déployer le package sur des instances gérées.

Rubriques

- Ajout d'un package (console) (p. 1191)
- Ajout d'un package (AWS CLI) (p. 1192)

Ajout d'un package (console)

Vous pouvez utiliser la console AWS Systems Manager pour créer un package. Vous devez avoir à votre disposition le nom du compartiment dans lequel vous avez chargé votre package dans Étape 3 : Charger le package et le manifeste dans un compartiment S3 (p. 1190).

Pour ajouter un package dans Distributor (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Distributor.
3. Sur la page d'accueil de Distributor, choisissez Create package (Créer un package), puis Advanced (Avancé).
4. Sur la page Create package (Créer un package), entrez un nom pour votre package. Les noms de package peuvent contenir des lettres, des chiffres, des points, des tirets et des traits de soulignement. Le nom doit être suffisamment générique pour pouvoir s'appliquer à toutes les versions des pièces jointes du package, mais suffisamment spécifique pour identifier l'objectif du package.
5. Pour Version name (Nom de la version), saisissez la valeur exacte de l'entrée `version` dans votre fichier manifeste.
6. Pour S3 bucket name (Nom du compartiment S3), choisissez le nom du compartiment dans lequel vous avez chargé vos fichiers .zip et le manifeste dans the section called “Étape 3 : Charger le package et le manifeste dans un compartiment S3” (p. 1190).

7. Dans S3 key prefix (Préfixe de clé S3), entrez le sous-dossier du compartiment dans lequel vos fichiers .zip et le manifeste sont stockés.
8. Dans Manifest (Manifeste), choisissez Extract from package (Extraire depuis le package) pour utiliser un manifeste que vous avez chargé dans le compartiment S3 avec vos fichiers .zip.

   (Facultatif) Si vous n'avez pas chargé votre manifeste JSON dans le compartiment S3 où vous avez stocké vos fichiers .zip, choisissez New manifest (Nouveau manifeste). Vous pouvez créer ou coller l'intégralité du champ de manifeste dans l'éditeur JSON. Pour de plus amples informations sur la création du manifeste JSON, veuillez consulter Étape 2 : Créer le manifeste du package JSON (p. 1185).
9. Lorsque vous avez fini avec le fichier manifeste, choisissez Create package (Créer un package).
10. Attendez que Distributor crée votre package à partir de vos fichiers .zip et de votre manifeste. Selon le nombre et la taille des packages que vous ajoutez, cela peut prendre quelques minutes. Distributor vous redirige vers la page Package details (Détails du package) pour le nouveau package, mais vous pouvez choisir d'ouvrir cette page vous-même une fois les logiciels chargés. La page Package details (Détails du package) n'affiche pas toutes les informations sur votre package tant que Distributor n'a pas terminé le processus de création de ce package. Pour arrêter le processus de chargement et de création de package, choisissez Cancel (Annuler).

Ajout d'un package (AWS CLI)

Vous pouvez utiliser l'AWS CLI pour créer un package. Vous devez avoir à votre disposition l'URL du compartiment dans lequel vous avez chargé votre package à l'Étape 3 : Charger le package et le manifeste dans un compartiment S3 (p. 1190).

Pour ajouter un package dans Amazon S3 (AWS CLI)

1. Pour utiliser l' AWS CLI afin de créer un package, exécutez la commande suivante, en remplaçant *`package-name`* par le nom de votre package et *`path-to-manifest-file`* avec le chemin d'accès de votre fichier manifeste JSON. *`DOC-EXAMPLE-BUCKET`* est l'URL du compartiment S3 dans lequel l'ensemble du package est stocké. Lorsque vous exécutez la commande create-document dans Distributor, vous devez spécifier la valeur `Package` pour `--document-type`.

   Si vous n'avez pas ajouté votre fichier manifeste au compartiment S3, la valeur du paramètre `--content` est le chemin d'accès au fichier manifeste JSON.

   ```
   aws ssm create-document \
       --name "package-name" \
       --content file://path-to-manifest-file \
       --attachments Key="SourceUrl",Values="DOC-EXAMPLE-BUCKET" \
       --version-name version-value-from-manifest \
       --document-type Package
   ```

   Voici un exemple pour .

   ```
   aws ssm create-document \
       --name "ExamplePackage" \
       --content file://path-to-manifest-file \
       --attachments Key="SourceUrl",Values="https://s3.amazonaws.com/DOC-EXAMPLE-BUCKET/
   ExamplePackage" \
       --version-name 1.0.1 \
       --document-type Package
   ```
2. Vérifiez que votre package a été ajouté et affichez le manifeste du package en exécutant la commande suivante, en remplaçant *`package-name`* par le nom de votre package. Pour obtenir une version spécifique du document (différente de la version d'un package), vous pouvez ajouter le paramètre `--document-version`.

```
aws ssm get-document \
    --name "package-name"
```

Pour de plus amples informations sur les autres options que vous pouvez utiliser avec la commande create-document, veuillez consulter create-document dans la section AWS Systems Manager du AWS CLI Command Reference. Pour de plus amples informations sur les autres options que vous pouvez utiliser avec la commande get-document, veuillez consulter get-document.

Modifier les autorisations du package (console)

Une fois que vous avez ajouté un package à AWS Systems Manager Distributor , vous pouvez modifier les autorisations du package dans la AWS Systems Manager console . Vous pouvez ajouter d'autres comptes AWS aux autorisations d'un package. Les packages peuvent être partagés avec d'autres comptes de la même région AWS uniquement. Le partage inter-régions n'est pas pris en charge. Par défaut, les packages sont définis sur Private (Privé), ce qui signifie que seules les personnes ayant accès au compte AWS du créateur du package peuvent afficher des informations sur le package, et mettre à jour ou supprimer ce dernier. Si les autorisations Private (Privé) sont acceptables, vous pouvez ignorer cette procédure.

Pour modifier les autorisations du package (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Distributor.
3. Dans la page Packages, choisissez le package dont vous souhaitez modifier les autorisations.
4. Sur l'onglet Package details (Détails du package), choisissez Edit permissions (Modifier les autorisations) pour pouvoir modifier les autorisations.
5. Pour Edit permissions (Modifier les autorisations), choisissez Shared with specific accounts (Partagé avec des comptes spécifiques).
6. Sous Shared with specific accounts (Partagé avec des comptes spécifiques), ajoutez les numéros de compte AWS, l'un après l'autre. Lorsque vous avez terminé, choisissez Save.

Modifier les balises du package (console)

Une fois que vous avez ajouté un package à AWS Systems Manager Distributor, vous pouvez modifier les balises du package dans la console AWS Systems Manager. Ces balises sont appliquées au package et ne sont pas liées aux balises des instances sur lesquelles vous souhaitez déployer le package. Les balises sont des paires clé-valeur sensibles à la casse qui peuvent vous aider à regrouper et filtrer vos packages en fonction des critères pertinents pour votre organisation. Si vous ne souhaitez pas ajouter de balises, vous êtes prêt à installer votre package ou à ajouter une nouvelle version.

Pour modifier les balises du package (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Distributor.
3. Dans la page Packages, choisissez le package dont vous souhaitez modifier les balises.
4. Sur l'onglet Package details (Détails du package), dans Tags (Balises), choisissez Edit (Modifier).
5. Pour Add tags (Ajouter des balises), entrez une clé de balise ou une paire clé de balise-valeur, puis choisissez Add (Ajouter). Répétez cette opération si vous souhaitez ajouter d'autres balises. Pour supprimer des balises, choisissez X sur la balise en bas de la fenêtre.

6. Une fois que vous avez fini d'ajouter les balises au package, choisissez Save (Enregistrer).

Ajouter une version de package à Distributor

Pour ajouter une version de package, créez un package (p. 1182), puis utilisez Distributor pour ajouter une version de package en ajoutant une entrée au document SSM qui existe déjà pour les anciennes versions. Pour gagner du temps, mettez à jour le manifeste d'une version antérieure du package, modifiez la valeur de l'entrée `version` dans le manifeste (par exemple, en remplaçant `Test_1.0` par `Test_2.0`) et enregistrez-le en tant que manifeste de la nouvelle version. Le flux de travail Add version (Ajouter une version) simple dans la console Distributor met à jour le fichier manifeste pour vous.

Une nouvelle version de package peut :

- Remplacer au moins l'un des fichiers installables attachés à la version actuelle.
- Ajouter de nouveaux fichiers installables pour prendre en charge d'autres plateformes.
- Supprimer des fichiers afin de stopper la prise en charge de plateformes spécifiques.

Une version plus récente peut utiliser le même compartiment S3, mais doit avoir une URL avec un autre nom de fichier affiché à la fin. Vous pouvez utiliser la console AWS Systems Manager ou l'AWS CLI pour ajouter la nouvelle version. Le chargement d'un fichier installable avec le nom exact d'un fichier installable existant dans le compartiment S3 remplace le fichier existant. Aucun fichier installable de la version antérieure n'est copié sur la nouvelle version ; vous devez charger les fichiers installables de la version antérieure pour qu'ils fassent partie d'une nouvelle version. Une fois que Distributor a fini de créer votre nouvelle version de package, vous pouvez supprimer ou réutiliser le compartiment S3, car Distributor copie vos logiciels dans un compartiment Systems Manager interne dans le cadre du processus de gestion des versions.

> **Note**
>
> Chaque package contient un maximum de 25 versions. Vous pouvez supprimer les versions qui ne sont plus requises.

Rubriques

- Ajout d'une version de package (console) (p. 1194)
- Ajout d'une version de package (AWS CLI) (p. 1197)

Ajout d'une version de package (console)

Avant d'effectuer ces étapes, suivez les instructions définies dans Créer un package (p. 1182) afin de créer un nouveau package pour la version. Utilisez ensuite la console AWS Systems Manager pour ajouter une nouvelle version de package à Distributor.

Ajout d'une version de package (simple)

Pour ajouter une version de package à l'aide du flux de travail Simple, préparez des fichiers installables mis à jour ou ajoutez des fichiers installables pour prendre en charge d'autres plateformes et architectures. Ensuite, utilisez Distributor pour charger les fichiers installables nouveaux et mis à jour, et ajouter une version de package. Le flux de travail Add version (Ajouter une version) simplifié dans la console Distributor met à jour le fichier manifeste et le document SSM associé pour vous.

Pour ajouter une version de package (simple)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.

2. Dans le panneau de navigation, choisissez Distributor.
3. Sur la page d'accueil de Distributor, choisissez le package auquel vous voulez ajouter une autre version.
4. Sur la page Add version (Ajouter une version), choisissez Simple.
5. Pour Version name (Nom de version), saisissez un nom de version. Le nom de version de la nouvelle version doit être différent des noms des anciennes versions. Les noms de version peuvent comporter 512 caractères au maximum et ne peuvent pas contenir de caractères spéciaux.
6. Pour S3 bucket name (Nom du compartiment S3), choisissez un compartiment S3 existant dans la liste. Il peut s'agir du même compartiment que celui que vous avez utilisé afin de stocker les fichiers installables pour les versions antérieures, mais les noms de fichier installable doivent être différents pour éviter d'écraser les fichiers installables existants dans le compartiment.
7. Pour S3 key prefix (Préfixe de clé S3), entrez le sous-dossier du compartiment dans lequel vos ressources installables sont stockées.
8. Pour Upload software (Charger des logiciels), recherchez les fichiers de logiciels installables que vous voulez attacher à la nouvelle version. Les fichiers installables de versions existantes ne sont pas automatiquement copiés vers une nouvelle version ; vous devez charger les fichiers installables de versions antérieures du package si vous voulez que les mêmes fichiers installables fassent partie de la nouvelle version. Vous pouvez charger plusieurs fichiers de logiciel en une seule action.
9. Pour Target platform (Plateforme cible), vérifiez que la plateforme de système d'exploitation cible pour chaque fichier installable est correcte. Si le système d'exploitation indiqué n'est pas correct, choisissez le système d'exploitation approprié dans la liste déroulante.

   Dans le flux de travail de gestion des versions Simple, comme vous chargez chaque fichier installable une seule fois, des étapes supplémentaires sont nécessaires pour cibler un seul fichier sur plusieurs systèmes d'exploitation. Par exemple, si vous chargez un fichier de logiciel installable nommé `Logtool_v1.1.1.rpm`, vous devez modifier certaines valeurs par défaut dans le flux de travail Simple pour demander à Distributor de cibler le même logiciel sur les systèmes d'exploitation Amazon Linux et Ubuntu. Vous pouvez effectuer l'une des actions suivantes pour contourner cette limite.

   - Utilisez plutôt le flux de travail de gestion des versions Advanced (Avancé), compressez chaque fichier installable en un fichier .zip avant de commencer et créez manuellement le manifeste afin qu'un fichier installable puisse cibler plusieurs plateformes ou versions de système d'exploitation. Pour plus d'informations, consultez la section Ajout d'une version de package (avancé) (p. 1196).
   - Modifiez manuellement le fichier manifeste dans le flux de travail Simple pour que votre fichier .zip cible plusieurs plateformes ou versions de système d'exploitation. Pour plus d'informations sur la façon de procéder, consultez la fin de l'étape 4 dans Étape 2 : Créer le manifeste du package JSON (p. 1185).
10. Pour la version de plateforme, vérifiez que la version de plateforme de système d'exploitation affichée est **`_any`** soit une version majeure suivie d'un caractère générique (7.*), soit la version exacte du système d'exploitation à laquelle vous souhaitez que votre logiciel s'applique. Pour plus d'informations sur la spécification d'une version de plateforme, consultez l'étape 4 de Étape 2 : Créer le manifeste du package JSON (p. 1185).
11. Pour Architecture, choisissez l'architecture de processeur correcte pour chaque fichier installable dans la liste déroulante. Pour plus d'informations sur les architectures prises en charge, consultez Architectures et plateformes de package prises en charge (p. 1176).
12. (Facultatif) Développez Scripts et vérifiez les scripts d'installation et de désinstallation générés par Distributor pour vos logiciels installables.
13. Pour ajouter d'autres fichiers de logiciels installables à la nouvelle version, choisissez Add software (Ajouter des logiciels). Sinon, accédez à l'étape suivante.
14. (Facultatif) Développez Manifest (Manifeste) et vérifiez le manifeste de package JSON généré par Distributor pour vos logiciels installables. Si vous avez modifié des informations relatives à vos logiciels installables depuis que vous avez commencé cette procédure, comme la version de plateforme ou la plateforme cible, choisissez Generate manifest (Générer un manifeste) pour afficher le manifeste de package mis à jour.

Vous pouvez modifier le manifeste manuellement si vous souhaitez cibler un logiciel installable pour plusieurs systèmes d'exploitation, comme décrit à l'étape 9. Pour plus d'informations sur la modification du manifeste, consultez Étape 2 : Créer le manifeste du package JSON (p. 1185).

15. Lorsque vous avez fini d'ajouter des logiciels et de vérifier les données de plateforme cible, de version et d'architecture, choisissez Add version (Ajouter une version).
16. Attendez que Distributor finisse de charger vos logiciels et de créer la nouvelle version de package. Distributor indique le statut de chargement pour chaque fichier installable. Selon le nombre et la taille des packages que vous ajoutez, cela peut prendre quelques minutes. Distributor vous redirige vers la page Package details (Détails du package) pour le package, mais vous pouvez choisir d'ouvrir cette page vous-même une fois les logiciels chargés. La page Package details (Détails du package) n'affiche pas toutes les informations sur votre package tant que Distributor n'a pas fini de créer la nouvelle version de package. Pour arrêter le chargement et la création de la version de package, choisissez Stop upload (Arrêter le chargement).
17. Si Distributor ne peut pas charger les fichiers de logiciels installables, il affiche un message Upload failed (Échec du chargement). Pour relancer le chargement, choisissez Retry upload (Réessayer le chargement). Pour plus d'informations sur la façon de résoudre les échecs de création de version de package, consultez Résolution des problèmes liés à AWS Systems Manager Distributor (p. 1209).
18. Lorsque Distributor a fini de créer la nouvelle version de package, sur la page Details (Détails) du package, dans l'onglet Versions, vous voyez s'afficher la nouvelle version dans la liste des versions de package disponibles. Définissez une version par défaut du package en choisissant une version, puis en choisissant Set default version (Définir la version par défaut).

    Si vous ne définissez aucune version par défaut, c'est la version la plus récente du package qui est la version par défaut.

Ajout d'une version de package (avancé)

Pour ajouter une version de package, créez un package (p. 1182), puis utilisez Distributor pour ajouter une version de package en ajoutant une entrée au document SSM qui existe déjà pour les anciennes versions. Pour gagner du temps, mettez à jour le manifeste d'une version antérieure du package, modifiez la valeur de l'entrée `version` dans le manifeste (par exemple, en remplaçant `Test_1.0` par `Test_2.0`) et enregistrez-le en tant que manifeste de la nouvelle version. Vous devez avoir un manifeste mis à jour pour ajouter une nouvelle version de package à l'aide du flux de travail Advanced (Avancé).

Pour ajouter une version de package (avancé)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Distributor.
3. Sur la page d'accueil de Distributor, choisissez le package auquel vous voulez ajouter une autre version, puis choisissez Ajouter une version.
4. Dans Version name (Nom de la version), saisissez la valeur exacte de l'entrée `version` de votre fichier manifeste.
5. Pour S3 bucket name (Nom du compartiment S3), choisissez un compartiment S3 existant dans la liste. Il peut s'agir du même compartiment que celui que vous avez utilisé afin de stocker les fichiers installables pour les versions antérieures, mais les noms de fichier installable doivent être différents pour éviter d'écraser les fichiers installables existants dans le compartiment.
6. Pour S3 key prefix (Préfixe de clé S3), entrez le sous-dossier du compartiment dans lequel vos ressources installables sont stockées.
7. Pour Manifest (Manifeste), choisissez Extract from package (Extraire depuis le package) pour utiliser un manifeste que vous avez chargé dans le compartiment S3 avec vos fichiers .zip.

   (Facultatif) Si vous n'avez pas chargé votre manifeste JSON révisé dans le compartiment S3 où vous avez stocké vos fichiers .zip, choisissez New manifest (Nouveau manifeste). Vous pouvez créer ou

coller l'intégralité du champ de manifeste dans l'éditeur JSON. Pour de plus amples informations sur la création du manifeste JSON, veuillez consulter Étape 2 : Créer le manifeste du package JSON (p. 1185).

8. Lorsque vous avez fini avec le fichier manifeste, choisissez Ajouter une version de package.
9. Sur la page Details (Détails) du package, dans l'onglet Versions, vous voyez s'afficher la nouvelle version dans la liste des versions de package disponibles. Définissez une version par défaut du package en choisissant une version, puis en choisissant Set default version (Définir la version par défaut).

   Si vous ne définissez aucune version par défaut, c'est la version la plus récente du package qui est la version par défaut.

Ajout d'une version de package (AWS CLI)

Vous pouvez utiliser l'AWS CLI pour ajouter une nouvelle version de package à Distributor. Avant d'exécuter ces commandes, vous devez créer une nouvelle version de package et la charger dans S3, comme décrit au début de cette rubrique.

Pour ajouter une version de package (AWS CLI)

1. Exécutez la commande suivante pour modifier le document AWS Systems Manager en ajoutant une entrée pour une nouvelle version de package. Remplacez *`document-name`* par le nom de votre document. – Remplacez *`DOC-EXAMPLE-BUCKET`* par l'URL du manifeste JSON que vous avez copié dans Étape 3 : Charger le package et le manifeste dans un compartiment S3 (p. 1190) . *`S3-bucket-URL-of-package`* est l'URL du compartiment S3 dans lequel l'ensemble du package est stocké. Remplacez *`version-name-from-updated-manifest`* avec la valeur de `version` dans le manifeste. Définissez le paramètre `--document-version` sur `$LATEST` pour que le document associé à cette version de package soit la dernière version du document.

```
aws ssm update-document \
    --name "document-name" \
    --content "S3-bucket-URL-to-manifest-file" \
    --attachments Key="SourceUrl",Values="DOC-EXAMPLE-BUCKET" \
    --version-name version-name-from-updated-manifest \
    --document-version $LATEST
```

   Voici un exemple pour .

```
aws ssm update-document \
    --name ExamplePackage \
    --content "https://s3.amazonaws.com/DOC-EXAMPLE-BUCKET/ExamplePackage/
manifest.json" \
    --attachments Key="SourceUrl",Values="https://s3.amazonaws.com/DOC-EXAMPLE-BUCKET/
ExamplePackage" \
    --version-name 1.1.1 \
    --document-version $LATEST
```

2. Exécutez la commande suivante pour vérifier que votre package a été mis à jour et afficher le manifeste du package. Remplacez *`package-name`* par le nom de votre package et éventuellement, *`document-version`* avec le numéro de version du document (pas le même que la version du package) que vous avez mis à jour. Si cette version de package est associée à la dernière version du document, vous pouvez spécifier `$LATEST` pour la valeur du paramètre facultatif `--document-version`.

```
aws ssm get-document \
    --name "package-name" \
    --document-version "document-version"
```

Pour de plus amples informations sur les autres options que vous pouvez utiliser avec la commande update-document, veuillez consulter update-document dans la section AWS Systems Manager du AWS CLI Command Reference.

Installer ou mettre à jour des packages

Vous pouvez utiliser AWS Management Console ou l'AWS CLI pour déployer des packages sur vos instances gérées AWS Systems Manager à l'aide d'AWS Systems Manager Distributor. Vous pouvez actuellement déployer une version d'un package par commande. Vous pouvez installer de nouveaux packages ou mettre à jour les installations existantes en place. Vous pouvez choisir de déployer une version spécifique ou choisir de toujours déployer la version la plus récente d'un package. Nous vous recommandons d'utiliser State Manager pour installer les packages. L'utilisation de State Manager permet de s'assurer que vos instances exécutent toujours la version la plus récente de votre package.

Préférence	Action AWS Systems Manager	En savoir plus
Installer ou mettre à jour un package immédiatement.	Fonctionnalité Exécuter la commande	• Installation ou mise à jour d'un package une fois (console) (p. 1199) • Installation unique d'un package (AWS CLI) (p. 1201) • Mise à jour unique d'un package (AWS CLI) (p. 1202)
Installer un package selon un calendrier, afin que l'installation inclue toujours la version par défaut.	State Manager	• Planification d'une installation ou d'une mise à jour de package (console) (p. 1200) • Planification d'une installation de package (AWS CLI) (p. 1202) • Planification d'une mise à jour de package (AWS CLI) (p. 1203)
Installer automatiquement un package sur de nouvelles instances ayant une balise spécifique ou un ensemble de balises spécifiques. Par exemple, installation de l'agent Amazon CloudWatch sur de nouvelles instances.	State Manager	Pour ce faire, vous pouvez appliquer des balises à de nouvelles instances, puis spécifier les balises en tant que cibles dans votre association State Manager. State Manager installe automatiquement le package dans une association sur les instances ayant des balises correspondantes. Veuillez consulter À propos des cibles et des contrôles du débit dans les associations State Manager (p. 990).

Rubriques

- Installation ou mise à jour d'un package une fois (console) (p. 1199)
- Planification d'une installation ou d'une mise à jour de package (console) (p. 1200)
- Installation unique d'un package (AWS CLI) (p. 1201)

- Mise à jour unique d'un package (AWS CLI) (p. 1202)
- Planification d'une installation de package (AWS CLI) (p. 1202)
- Planification d'une mise à jour de package (AWS CLI) (p. 1203)

Installation ou mise à jour d'un package une fois (console)

Vous pouvez utiliser la console AWS Systems Manager pour effectuer une installation ou une mise à jour unique d'un package. Lorsque vous configurez une installation unique, Distributor utilise AWS Systems Manager Fonctionnalité Exécuter la commande (p. 941) pour effectuer l'installation.

Pour installer ou mettre à jour un package une fois (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Distributor.
3. Dans la page d'accueil de Distributor, choisissez le package à installer.
4. Choisissez Installer une fois.

   Cette commande ouvre Systems Manager Fonctionnalité Exécuter la commande avec le document de commande `AWS-ConfigureAWSPackage` et votre package Distributor déjà sélectionné.
5. Pour Document version (Version du document), sélectionnez la version du document `AWS-ConfigureAWSPackage` à exécuter.
6. Pour Actions, choisissez Installer.
7. Pour Installation type (Type d'installation), choisissez l'une des valeurs suivantes :
   - Désinstaller et réinstaller : Le package est complètement désinstallé, puis réinstallé. L'application n'est pas disponible tant que la réinstallation n'est pas terminée.
   - Mise à jour sur place : Seuls les fichiers nouveaux ou modifiés sont ajoutés à l'installation existante selon les instructions que vous fournissez dans un `update` script . L'application reste disponible tout au long du processus de mise à jour. Cette option n'est pas prise en charge pour les packages publiés par AWS.
8. Pour Name (Nom), vérifiez que le nom du package sélectionné est entré.
9. Pour Version, entrez la valeur du nom de version du package. Si vous laissez ce champ vide, Fonctionnalité Exécuter la commande installe la version par défaut que vous avez sélectionnée dans Distributor.
10. Dans la section Cibles, identifiez les instances sur lesquelles vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant manuellement les instances ou en indiquant un groupe de ressources.

    > **Note**
    >
    > Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.
11. Pour Autres paramètres :
    - Pour Comment (Commentaire), saisissez des informations à propos de cette commande.
    - Pour Délai (secondes), précisez le nombre de secondes durant lesquelles le système doit attendre avant de mettre en échec l'exécution de la commande globale.
12. (Facultatif) Pour Contrôle du débit :
    - Pour Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage d'instances sur lesquelles exécuter simultanément la commande.

Note

Si vous avez sélectionné des cibles en indiquant des balises appliquées à des instances gérées ou en indiquant des groupes de ressources AWS, et que vous n'êtes pas sûr du nombre d'instances ciblées, limitez le nombre d'instances pouvant exécuter simultanément le document en spécifiant un pourcentage.

- Pour Seuil d'erreur, spécifiez quand arrêter l'exécution de la commande sur d'autres instances après son échec sur un nombre ou un pourcentage d'instances. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les instances sur lesquelles la commande est en cours de traitement peuvent également envoyer des erreurs.

13. (Facultatif) Pour Output options (Options de sortie), pour enregistrer la sortie de la commande dans un fichier, cochez la case Write command output to an S3 bucket (Écrire la sortie de commande vers un compartiment S3). Entrez les noms de compartiment et de préfixe (dossier) dans les zones.

    Note

    Les autorisations S3 qui accordent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour de plus amples informations, veuillez consulter Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). En outre, si le compartiment S3 spécifié se trouve dans un compte AWS différent, assurez-vous que le profil d'instance associé à l'instance dispose des autorisations nécessaires pour écrire dans ce compartiment.
14. Dans la section SNS notifications (Notifications SNS), si vous souhaitez que des notifications sur le statut de l'exécution de la commande soient envoyées, cochez la case Enable SNS notifications (Activer les notifications SNS).

    Pour plus d'informations sur la configuration des notifications Amazon SNS pour la fonctionnalité Exécuter la commande, consultez Surveillance des changements de statut Systems Manager à l'aide de notifications Amazon SNS (p. 1344).
15. Lorsque vous êtes prêt à installer le package, choisissez Run (Exécuter).
16. La zone Command status (État de la commande) indique la progression de l'exécution. Si la commande est toujours en cours, choisissez l'icône d'actualisation dans l'angle supérieur gauche de la console jusqu'à ce que la colonne Overall status (État général) ou Detailed status (État détaillé) affiche Success (Succès) ou Failure (Échec).
17. Dans la zone Targets and outputs (Cibles et sorties) cliquez sur le bouton en regard d'un nom d'instance, puis choisissez View output (Afficher la sortie).

    La page de sortie de la commande illustre les résultats de l'exécution de votre commande.
18. (Facultatif) Si vous avez choisi d'écrire la sortie de commande dans un compartiment S3, choisissez Amazon S3 pour afficher les données du journal de sortie.

Planification d'une installation ou d'une mise à jour de package (console)

Vous pouvez utiliser la console AWS Systems Manager pour planifier l'installation ou la mise à jour d'un package. Lorsque vous planifiez l'installation ou la mise à jour du package, Distributor utilise AWS Systems Manager State Manager (p. 987) pour effectuer l'installation ou la mise à jour.

Pour planifier une installation de package (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Distributor.
3. Dans la page d'accueil de Distributor, choisissez le package à installer ou mettre à jour.
4. Pour Package, choisissez Install on a schedule (Installation planifiée).

Cette commande ouvre Systems Manager State Manager dans une nouvelle association qui est créée pour vous.

5. Pour Name (Nom), saisissez un nom (par exemple, **`Deploy-test-agent-package`**). Cette action est facultative, mais recommandée. Les espaces ne sont pas autorisés dans le nom.
6. Dans la liste Document, le nom du document `AWS-ConfigureAWSPackage` est déjà sélectionné.
7. Pour Action, vérifiez que Installer est sélectionné.
8. Pour Installation type (Type d'installation), choisissez l'une des valeurs suivantes :
   - Désinstaller et réinstaller : Le package est complètement désinstallé, puis réinstallé. L'application n'est pas disponible tant que la réinstallation n'est pas terminée.
   - Mise à jour sur place : Seuls les fichiers nouveaux ou modifiés sont ajoutés à l'installation existante selon les instructions que vous fournissez dans un `update` script . L'application reste disponible tout au long du processus de mise à jour.
9. Pour Name (Nom), vérifiez que le nom de votre package est entré.
10. Pour Version, si vous souhaitez installer une version de package autre que la dernière version publiée, entrez l'identificateur de version.
11. Pour Targets (Cibles), choisissez Selecting all managed instances in this account (Sélection de toutes les instances gérées dans ce compte), Specifying tags (Spécification des balises) ou Manually Selecting Instance (Sélection manuelle des instances). Si vous choisissez de cibler des ressources à l'aide de balises, entrez une clé de balise et une valeur de balise dans les champs fournis.
12. Pour Specify schedule (Spécifier le calendrier), choisissez On Schedule (Selon le calendrier) pour exécuter l'association selon une planification régulière ou No Schedule (Pas de calendrier) pour exécuter l'association une seule fois. Pour de plus amples informations sur ces options, veuillez consulter Créer une association (p. 993). Utilisez les contrôles pour créer un calendrier de type `cron` ou rate pour l'association.
13. Choisissez Create Association (Créer une association).
14. Sur la page Association cliquez sur le bouton en regard de l'association que vous avez créée, puis choisissez Apply association now (Appliquer l'association maintenant).

    State Manager crée et exécute immédiatement l'association sur les instances ou les cibles spécifiées. Pour de plus amples informations sur les résultats de l'exécution des associations, veuillez consulter Créer une association (p. 993) dans ce guide.

Pour de plus amples informations sur l'utilisation des options dans Advanced options (Options avancées), Rate control (Contrôle de débit) et Output options (Options de sortie), veuillez consulter Créer une association (p. 993).

Installation unique d'un package (AWS CLI)

Vous pouvez exécuter send-command dans l'AWS CLI pour effectuer une installation unique d'un package Distributor. Si le package est déjà installé, l'application sera déconnectée pendant que le package est désinstallé et que la nouvelle version est installée à sa place.

Pour effectuer une installation unique d'un package (AWS CLI)

- Exécutez la commande suivante dans l'AWS CLI :

```
aws ssm send-command \
    --document-name "AWS-ConfigureAWSPackage" \
    --instance-ids "instance-IDs" \
    --parameters '{"action":["Install"],"installationType":["Uninstall and
 reinstall"],"name":["package-name (in same account) or package-ARN (shared from
 different account)"]}'
```

> **Note**
>
> Le comportement par défaut pour `installationType` est `Uninstall and reinstall` . Vous pouvez omettre `"installationType":["Uninstall and reinstall"]` de cette commande lorsque vous installez un package complet.

Voici un exemple pour .

```
aws ssm send-command \
    --document-name "AWS-ConfigureAWSPackage" \
    --instance-ids "i-00000000000000" \
    --parameters '{"action":["Install"],"installationType":["Uninstall and
 reinstall"],"name":["ExamplePackage"]}'
```

Pour de plus amples informations sur les autres options que vous pouvez utiliser avec la commande send-command, veuillez consulter send-command dans la section AWS Systems Manager du AWS CLI Command Reference.

Mise à jour unique d'un package (AWS CLI)

Vous pouvez exécuter send-command dans l'AWS CLI pour mettre à jour un package Distributor sans mettre l'application associée hors connexion. Seuls les fichiers nouveaux ou mis à jour dans le package sont remplacés.

Pour effectuer une mise à jour unique d'un package (AWS CLI)

- Exécutez la commande suivante dans l'AWS CLI :

```
aws ssm send-command \
    --document-name "AWS-ConfigureAWSPackage" \
    --instance-ids "instance-IDs" \
    --parameters '{"action":["Install"],"installationType":["In-place update"],"name":
["package-name (in same account) or package-ARN (shared from different account)"]}'
```

> **Note**
>
> Lorsque vous ajoutez des fichiers nouveaux ou modifiés, vous devez inclure `"installationType":["In-place update"]` dans la commande.

Voici un exemple pour .

```
aws ssm send-command \
    --document-name "AWS-ConfigureAWSPackage" \
    --instance-ids "i-02573cafcfEXAMPLE" \
    --parameters '{"action":["Install"],"installationType":["In-place update"],"name":
["ExamplePackage"]}'
```

Pour de plus amples informations sur les autres options que vous pouvez utiliser avec la commande send-command, veuillez consulter send-command dans la section AWS Systems Manager du AWS CLI Command Reference.

Planification d'une installation de package (AWS CLI)

Vous pouvez exécuter create-association dans l'AWS CLI pour installer un package Distributor selon un calendrier. La valeur de `--name` , le nom du document, est toujours `AWS-ConfigureAWSPackage` . La

commande suivante utilise la clé `InstanceIds` pour spécifier les instances cibles. Si le package est déjà installé, l'application sera déconnectée pendant que le package est désinstallé et que la nouvelle version est installée à sa place.

```
aws ssm create-association \
    --name "AWS-ConfigureAWSPackage" \
    --parameters '{"action":["Install"],"installationType":["Uninstall and
 reinstall"],"name":["package-name (in same account) or package-ARN (shared from different
 account)"]}' \
    --targets [{\"Key\":\"InstanceIds\",\"Values\":[\"instance-ID1\",\"instance-ID2\"]}]
```

> **Note**
>
> Le comportement par défaut pour `installationType` est `Uninstall and reinstall` . Vous pouvez omettre `"installationType":["Uninstall and reinstall"]` de cette commande lorsque vous installez un package complet.

Voici un exemple pour .

```
aws ssm create-association \
    --name "AWS-ConfigureAWSPackage" \
    --parameters '{"action":["Install"],"installationType":["Uninstall and
 reinstall"],"name":["Test-ConfigureAWSPackage"]}' \
    --targets [{\"Key\":\"InstanceIds\",\"Values\":[\"i-02573cafcfEXAMPLE\",
\"i-0471e04240EXAMPLE\"]}]
```

Pour de plus amples informations sur les autres options que vous pouvez utiliser avec la commande create-association, veuillez consulter create-association dans la section AWS Systems Manager du AWS CLI Command Reference.

Planification d'une mise à jour de package (AWS CLI)

Vous pouvez exécuter create-association dans l'AWS CLI pour mettre à jour un package Distributor selon une planification sans mettre l'application associée hors connexion. Seuls les fichiers nouveaux ou mis à jour dans le package sont remplacés. La valeur de `--name` , le nom du document, est toujours `AWS-ConfigureAWSPackage` . La commande suivante utilise la clé `InstanceIds` pour spécifier les instances cibles.

```
aws ssm create-association \
    --name "AWS-ConfigureAWSPackage" \
    --parameters '{"action":["Install"],"installationType":["In-place update"],"name":
["package-name (in same account) or package-ARN (shared from different account)"]}' \
    --targets [{\"Key\":\"InstanceIds\",\"Values\":[\"instance-ID1\",\"instance-ID2\"]}]
```

> **Note**
>
> Lorsque vous ajoutez des fichiers nouveaux ou modifiés, vous devez inclure `"installationType":["In-place update"]` dans la commande.

Voici un exemple pour .

```
aws ssm create-association \
    --name "AWS-ConfigureAWSPackage" \
    --parameters '{"action":["Install"],"installationType":["In-place update"],"name":
["Test-ConfigureAWSPackage"]}' \
    --targets [{\"Key\":\"InstanceIds\",\"Values\":[\"i-02573cafcfEXAMPLE\",
\"i-0471e04240EXAMPLE\"]}]
```

Pour de plus amples informations sur les autres options que vous pouvez utiliser avec la commande create-association, veuillez consulter create-association dans la section AWS Systems Manager du AWS CLI Command Reference.

Désinstaller un package

Vous pouvez utiliser AWS Management Console ou l'AWS CLI pour désinstaller des packages Distributor de vos instances gérées AWS Systems Manager à l'aide de Fonctionnalité Exécuter la commande. Dans cette version, vous pouvez désinstaller une version d'un package par commande. Vous pouvez désinstaller une version spécifique ou la version par défaut.

Rubriques

- Désinstallation d'un package (console) (p. 1204)
- Désinstallation d'un package (AWS CLI) (p. 1204)

Désinstallation d'un package (console)

Vous pouvez utiliser Fonctionnalité Exécuter la commande dans la console AWS Systems Manager pour désinstaller un package ponctuellement. Distributor utilise AWS Systems Manager Fonctionnalité Exécuter la commande (p. 941) pour désinstaller les packages.

Pour désinstaller un package (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fonctionnalité Exécuter la commande.
3. Sur la page d'accueil de Fonctionnalité Exécuter la commande, choisissez Run command (Exécuter la commande).
4. Choisissez le document de commande `AWS-ConfigureAWSPackage`.
5. Dans Action, choisissez Uninstall (Désinstaller)
6. Pour Name (Nom), tapez le nom du package à désinstaller.
7. Dans Targets (Cibles), choisissez les instances gérées cible en spécifiant une clé de balise et les valeurs qui sont partagées par les instances gérées, ou spécifiez les instances à l'aide de l'un des sept attributs, en particulier l'ID d'instance, la plateforme et la version de l'Agent SSM.
8. Vous pouvez utiliser les options avancées pour ajouter des commentaires sur l'opération, modifier les valeurs Concurrency (Simultanéité) et Error threshold (Seuil d'erreur) dans Rate control (Contrôle de débit), spécifier des options de sortie ou configurer les notifications Amazon SNS. Pour plus d'informations, consultez Exécution des commandes depuis la console dans ce guide.
9. Lorsque vous êtes prêt à désinstaller le package, choisissez Run (Exécuter), puis View results (Afficher les résultats).
10. Dans la liste des commandes, choisissez la commande `AWS-ConfigureAWSPackage` que vous venez d'exécuter. Si la commande est toujours en cours d'exécution, choisissez l'icône d'actualisation dans le coin supérieur droit de la console.
11. Quand la colonne Status (Statut) affiche Success (Réussite) ou Failed (Échec), choisissez l'onglet Output (Sortie).
12. Choisissez View output (Afficher la sortie). La page de sortie de la commande illustre les résultats de l'exécution de votre commande.

Désinstallation d'un package (AWS CLI)

Vous pouvez utiliser l'AWS CLI pour désinstaller un package Distributor de vos instances gérées à l'aide de Fonctionnalité Exécuter la commande.

Pour désinstaller un package (AWS CLI)

- Exécutez la commande suivante dans l'AWS CLI :

```
aws ssm send-command \
    --document-name "AWS-ConfigureAWSPackage" \
    --instance-ids "instance-IDs" \
    --parameters '{"action":["Uninstall"],"name":["package-name (in same account) or
 package-ARN (shared from different account)"]}'
```

Voici un exemple pour .

```
aws ssm send-command \
    --document-name "AWS-ConfigureAWSPackage" \
    --instance-ids "i-02573cafcfEXAMPLE" \
    --parameters '{"action":["Uninstall"],"name":["Test-ConfigureAWSPackage"]}'
```

Pour de plus amples informations sur les autres options que vous pouvez utiliser avec la commande send-command, veuillez consulter send-command dans la section AWS Systems Manager du AWS CLI Command Reference.

Supprimer un package

Suppression d'un package (console)

Vous pouvez utiliser la console AWS Systems Manager pour supprimer un package de Distributor. La suppression d'un package supprime toutes les versions du package de Distributor.

Pour supprimer un package (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Distributor.
3. Dans la page d'accueil de Distributor, choisissez le package à supprimer.
4. Sur la page des détails du package, choisissez Delete package (Supprimer le package).
5. Lorsque vous êtes invité à confirmer la suppression, choisissez Delete package (Supprimer le package).

Suppression d'une version de package (console)

Vous pouvez utiliser la console AWS Systems Manager pour supprimer une version de package de Distributor.

Pour supprimer une version de package (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Distributor.
3. Sur la page d'accueil de Distributor, choisissez le package dont vous souhaitez supprimer une version.
4. Sur la page des versions du package, choisissez la version à supprimer et choisissez Delete version (Supprimer la version).
5. Lorsque vous êtes invité à confirmer la suppression, choisissez Delete package version (Supprimer la version de package).

Suppression d'un package (ligne de commande)

Vous pouvez utiliser votre outil de ligne de commande préféré pour supprimer un package de Distributor.

Linux

Pour supprimer un package (AWS CLI)

1. Exécutez la commande suivante pour répertorier les documents de packages spécifiques. Dans les résultats de cette commande, recherchez le package que vous souhaitez supprimer.

```
aws ssm list-documents \
    --filters Key=Name,Values=package-name
```

2. Exécutez la commande suivante pour supprimer un package. Remplacez *package-name* par le nom du package.

```
aws ssm delete-document \
    --name "package-name"
```

3. Exécutez à nouveau la commande list-documents pour vérifier que le package a été supprimé. Le package que vous avez supprimé ne doit pas apparaître dans la liste.

```
aws ssm list-documents \
    --filters Key=Name,Values=package-name
```

Windows

Pour supprimer un package (AWS CLI)

1. Exécutez la commande suivante pour répertorier les documents de packages spécifiques. Dans les résultats de cette commande, recherchez le package que vous souhaitez supprimer.

```
aws ssm list-documents ^
    --filters Key=Name,Values=package-name
```

2. Exécutez la commande suivante pour supprimer un package. Remplacez *package-name* par le nom du package.

```
aws ssm delete-document ^
    --name "package-name"
```

3. Exécutez à nouveau la commande list-documents pour vérifier que le package a été supprimé. Le package que vous avez supprimé ne doit pas apparaître dans la liste.

```
aws ssm list-documents ^
    --filters Key=Name,Values=package-name
```

PowerShell

Pour supprimer un package (Outils pour PowerShell)

1. Exécutez la commande suivante pour répertorier les documents de packages spécifiques. Dans les résultats de cette commande, recherchez le package que vous souhaitez supprimer.

```
$filter = New-Object Amazon.SimpleSystemsManagement.Model.DocumentKeyValuesFilter
```

```
$filter.Key = "Name"
$filter.Values = "package-name"

Get-SSMDocumentList `
    -Filters @($filter)
```

2. Exécutez la commande suivante pour supprimer un package. Remplacez *package-name* par le nom du package.

```
Remove-SSMDocument `
    -Name "package-name"
```

3. Exécutez à nouveau la commande Get-SSMDocumentList pour vérifier que le package a été supprimé. Le package que vous avez supprimé ne doit pas apparaître dans la liste.

```
$filter = New-Object Amazon.SimpleSystemsManagement.Model.DocumentKeyValuesFilter
$filter.Key = "Name"
$filter.Values = "package-name"

Get-SSMDocumentList `
    -Filters @($filter)
```

Suppression d'une version de package (ligne de commande)

Vous pouvez utiliser votre outil de ligne de commande préféré pour supprimer une version de package de Distributor.

Linux

Pour supprimer une version de package (AWS CLI)

1. Exécutez la commande suivante pour répertorier les versions de votre package. Dans les résultats de cette commande, recherchez la version de package que vous souhaitez supprimer.

```
aws ssm list-document-versions \
    --name "package-name"
```

2. Exécutez la commande suivante pour supprimer une version de package. Remplacez *package-name* par le nom du package et *version* par le numéro de version.

```
aws ssm delete-document \
    --name "package-name" \
    --document-version version
```

3. Exécutez la commande list-document-versions pour vérifier que la version du package a été supprimée. La version de package que vous avez supprimée doit être introuvable.

```
aws ssm list-document-versions \
    --name "package-name"
```

Windows

Pour supprimer une version de package (AWS CLI)

1. Exécutez la commande suivante pour répertorier les versions de votre package. Dans les résultats de cette commande, recherchez la version de package que vous souhaitez supprimer.

```
aws ssm list-document-versions ^
    --name "package-name"
```

2. Exécutez la commande suivante pour supprimer une version de package. Remplacez *package-name* par le nom du package et *version* par le numéro de version.

```
aws ssm delete-document ^
    --name "package-name" ^
    --document-version version
```

3. Exécutez la commande list-document-versions pour vérifier que la version du package a été supprimée. La version de package que vous avez supprimée doit être introuvable.

```
aws ssm list-document-versions ^
    --name "package-name"
```

PowerShell

Pour supprimer une version de package (Outils pour PowerShell)

1. Exécutez la commande suivante pour répertorier les versions de votre package. Dans les résultats de cette commande, recherchez la version de package que vous souhaitez supprimer.

```
Get-SSMDocumentVersionList `
    -Name "package-name"
```

2. Exécutez la commande suivante pour supprimer une version de package. Remplacez *package-name* par le nom du package et *version* par le numéro de version.

```
Remove-SSMDocument `
    -Name "package-name" `
    -DocumentVersion version
```

3. Exécutez la commande Get-SSMDocumentVersionList pour vérifier que la version du package a été supprimée. La version de package que vous avez supprimée doit être introuvable.

```
Get-SSMDocumentVersionList `
    -Name "package-name"
```

Pour de plus amples informations sur les autres options que vous pouvez utiliser avec la commande list-documents, veuillez consulter list-documents dans la section AWS Systems Manager du AWS CLI Command Reference. Pour de plus amples informations sur les autres options que vous pouvez utiliser avec la commande delete-document, veuillez consulter delete-document.

Audit et journalisation de l'activité de Distributor

Pour de plus amples informations sur les options d'audit et de journalisation d'AWS Systems Manager, veuillez consulter Surveillance des AWS Systems Manager (p. 1326).

Audit de l'activité de Distributor avec AWS CloudTrail

AWS CloudTrail capture les appels d'API effectués dans la console Systems Manager, l'AWS CLI et le kit SDK Systems Manager. Les informations peuvent être affichées sur la console CloudTrail ou stockées

dans un compartiment S3. Un seul compartiment est utilisé pour tous les journaux CloudTrail de votre compte.

Les journaux des actions Fonctionnalité Exécuter la commande et State Manager présentent les activités de création de documents, d'installation et de désinstallation de packages. Pour de plus amples informations sur l'affichage et l'utilisation des journaux CloudTrail de l'activité Systems Manager, veuillez consulter Journalisation des appels d'API AWS Systems Manager avec AWS CloudTrail (p. 1336).

Résolution des problèmes liés à AWS Systems Manager Distributor

Les informations suivantes peuvent vous aider à résoudre les problèmes liés à l'utilisation de Distributor.

Rubriques

- Un package erroné portant le même nom est installé (p. 1209)
- Erreur : Échec de la récupération du manifeste : Impossible de trouver la dernière version du package (p. 1209)
- Erreur : Échec de la récupération du manifeste : Exception de validation (p. 1209)

Un package erroné portant le même nom est installé

Problème : Vous avez installé un package, mais AWS Systems Manager Distributor vous avez installé un autre package à la place.

Cause: Pendant l'installation, AWS Systems Manager recherche les packages publiés par avant les packages externes définis par l'utilisateur.AWS Si le nom du package défini par l'utilisateur est identique à celui d'un package publié par AWS, c'est le package AWS qui est installé à la place du vôtre.

Solution: Pour éviter ce problème, nommez votre package avec un autre nom que celui d'un package publié par .AWS

Erreur : Échec de la récupération du manifeste : Impossible de trouver la dernière version du package

Problème : Vous avez reçu une erreur similaire à la suivante :

```
Failed to retrieve manifest: ResourceNotFoundException: Could not find the latest version
 of package
arn:aws:ssm:::package/package-name status code: 400, request id: guid
```

Cause: Vous utilisez une version d' Agent SSM avec Systems Manager Distributor qui est antérieure à la version 2.3.274.0.

Solution: Mettez à jour la version de Agent SSM vers la version 2.3.274.0 ou ultérieure. Pour de plus amples informations, veuillez consulter Mettre à jour l'Agent SSM à l'aide de Fonctionnalité Exécuter la commande (p. 948) ou Mise à jour automatique d'Agent SSM (interface de ligne de commande) (p. 1043).

Erreur : Échec de la récupération du manifeste : Exception de validation

Problème : Vous avez reçu une erreur similaire à la suivante :

```
Failed to retrieve manifest: ValidationException: 1 validation error detected: Value
 'documentArn'
at 'packageName' failed to satisfy constraint: Member must satisfy regular expression
 pattern:
arn:aws:ssm:region-id:account-id:package/package-name
```

Cause: Vous utilisez une version d' Agent SSM avec Systems Manager Distributor qui est antérieure à la version 2.3.274.0.

Solution: Mettez à jour la version de Agent SSM vers la version 2.3.274.0 ou ultérieure. Pour de plus amples informations, veuillez consulter Mettre à jour l'Agent SSM à l'aide de Fonctionnalité Exécuter la commande (p. 948) ou Mise à jour automatique d'Agent SSM (interface de ligne de commande) (p. 1043).

Ressources AWS Systems Manager partagées

Systems Manager utilise les ressources partagées suivantes pour gérer et configurer vos ressources AWS.

Rubriques

- Documents AWS Systems Manager (p. 1211)

Documents AWS Systems Manager

Un AWS Systems Manager document (SSMdocument) définit les actions qu' Systems Manager effectue sur vos instances gérées. Systems Manager inclut plus de 100 documents préconfigurés que vous pouvez utiliser en spécifiant des paramètres lors de l'exécution. Les documents utilisent JavaScript le format JSON (Object Notation) ou YAML et incluent les étapes et les paramètres que vous spécifiez.

Types de documents SSM

Le tableau suivant décrit les différents types de documents SSM.

Tapez	A utiliser avec	Details (Détails)
Document de commande	Fonctionnalité Exécuter la commande (p. 941) State Manager (p. 987) Fenêtres de maintenance (p. 688)	Fonctionnalité Exécuter la commande utilise des documents de commande pour exécuter des commandes. State Manager utilise des documents de commande pour appliquer une configuration. Ces actions peuvent être exécutées sur une ou plusieurs cibles lors du cycle de vie d'une instance. Fenêtres de maintenance utilise des documents de commande pour appliquer une configuration en fonction de la planification spécifiée.
Document Automation	Automation (p. 304) State Manager (p. 987) Fenêtres de maintenance (p. 688)	Utilisez des documents Automation lors de l'exécution de tâches de maintenance et de déploiement courantes, telles que la création ou la mise à jour d'une Amazon Machine Image (AMI). State Manager utilise des documents Automation pour appliquer une configuration. Ces actions peuvent être exécutées sur une ou plusieurs cibles lors du cycle de vie d'une instance. Fenêtres de maintenance utilise des documents Automation pour

Tapez	A utiliser avec	Details (Détails)
		effectuer les tâches courantes de maintenance et de déploiement en fonction de la planification spécifiée.
Document de package	Distributor (p. 1174)	Dans Distributor, un package est représenté par un document SSM. Un document de package inclut des fichiers d'archive ZIP attachés qui contiennent des logiciels ou des ressources à installer sur les instances gérées. La création d'un package dans Distributor crée le document de package.
Document de session	Session Manager (p. 869)	Session Manager utilise des documents de session pour déterminer le type de session à démarrer, telle qu'une session de réacheminement de port, une session d'exécution de commande interactive ou une session de création de tunnel SSH.
Document de stratégie	State Manager (p. 987)	Systems Manager Inventory utilise le document de stratégie AWS-GatherSoftwareInventory avec une association State Manager pour collecter les données d'inventaire à partir des instances gérées. Lors de la création de vos propres documents SSM, les documents Automation et les documents Run Command constituent la méthode privilégiée pour appliquer une stratégie à une instance gérée.
Document Change Calendar	Modifier le calendrier (p. 680)	Systems Manager Modifier le calendrier utilise le type de document `ChangeCalendar`. Un document Modifier le calendrier stocke une entrée de calendrier et les événements associés qui peuvent autoriser ou empêcher les actions Automation de modifier votre environnement. Dans Modifier le calendrier , un document stocke les données iCalendar 2.0 au format texte brut.

Versions et exécution des documents SSM

Vous pouvez créer et enregistrer différentes versions des documents. Vous pouvez ensuite spécifier une version par défaut pour chaque document. La version par défaut d'un document peut être mise à jour vers une version plus récente ou rétablie vers une version plus ancienne du document. Lorsque vous modifiez le contenu d'un document, Systems Manager incrémente automatiquement la version du document. Vous pouvez récupérer ou utiliser n'importe quelle version d'un document en spécifiant la version du document dans la console, les commandes CLI ou les appels d'API.

Personnalisation d'un document

Si vous souhaitez personnaliser les étapes et les actions dans un document, vous pouvez créer votre propre document. La première fois que vous créez utiliser un document pour exécuter une action sur une instance, le système stocke ce document sur votre compte utilisateur AWS. Pour de plus amples informations sur la création d'un document SSM, veuillez consulter Création de documents Systems Manager (p. 1267).

Balisage d'un document

Vous pouvez baliser vos documents pour vous aider à identifier rapidement un ou plusieurs documents en fonction des balises que vous lui avez affectées. Par exemple, vous pouvez baliser des documents pour des environnements, des services, des utilisateurs, des groupes ou des périodes spécifiques. Vous pouvez également restreindre l'accès aux documents en créant une stratégie IAM qui spécifie les balises auxquelles un utilisateur ou un groupe peuvent accéder. Pour plus d'informations, consultez la section Balisage de documents Systems Manager (p. 1360).

Partage d'un document

Vous pouvez rendre vos documents publics ou les partager avec des comptes AWS spécifiques de la même région AWS. Le partage des documents entre plusieurs comptes peut être utile si, par exemple, vous souhaitez que toutes les instances EC2 que vous fournissez aux clients ou aux employés aient la même configuration. En plus de maintenir des applications ou des correctifs sur les instances à jour, il se peut que vous souhaitiez limiter les instances du client à partir de certaines activités. Il est également possible de s'assurer que les employés utilisés par des comptes au sein de toute votre organisation aient accès à certaines ressources internes. Pour plus d'informations, consultez la section Partage de documents SSM (p. 1275).

Quotas de documents SSM

Pour plus d'informations sur les quotas de documents SSM, consultez Quotas de service Systems Manager dans le Référence générale d'Amazon Web Services..

Rubriques

- Schémas et fonctionnalités de document SSM (p. 1213)
- Syntaxe du document SSM (p. 1226)
- Référence du plug-in de document Systems Manager (p. 1232)
- Création de documents Systems Manager (p. 1267)
- Partage de documents SSM (p. 1275)
- Recherche de SSM documents (p. 1283)
- Exécution de documents Systems Manager de commande à partir d'emplacements distants (p. 1286)

Schémas et fonctionnalités de document SSM

Les documents SSM utilisent actuellement les versions de schéma ci-dessous.

- Les documents de type `Command` peuvent utiliser la version de schéma 1.2, 2.0, et 2.2. Si vous utilisez actuellement des documents de schéma 1.2, nous vous recommandons de créer des documents qui utilisent la version de schéma 2.2.
- Les documents de type `Policy` doivent utiliser la version de schéma 2.0 ou ultérieure.
- Les documents de type `Automation` doivent utiliser la version de schéma 0.3.
- Vous pouvez créer des documents au format JSON ou YAML.

En utilisant la dernière version de schéma pour les documents `Command` et `Policy`, vous pouvez profiter des fonctions suivantes.

Fonctionnalités d'un document de version de schéma 2.2

Fonctionnalité	Details (Détails)
Modification du document	Les documents peuvent désormais être mis à jour. Avec la version 1.2, la mise à jour d'un document nécessitait qu'il soit enregistré sous un autre nom.
Gestion automatique des versions	Toute mise à jour d'un document crée une nouvelle version. Il ne s'agit pas d'une version de schéma, mais d'une version du document.
Version par défaut	Si vous disposez de plusieurs versions d'un document, vous pouvez spécifier la version qui est le document par défaut.
Séquençage	Les plug-ins ou étapes dans un document s'exécutent dans l'ordre que vous avez spécifié.
Support multiplateforme	Le support multiplateforme vous permet de spécifier un système d'exploitation différent pour différents plug-ins dans le même document SSM. Le support multiplateforme utilise le même paramètre `precondition` dans une étape.

> **Note**
>
> Vous devez conserver l'Agent SSM sur vos instances à jour avec la dernière version pour utiliser les nouvelles fonctions Systems Manager ainsi que les fonctions de document SSM. Pour plus d'informations, consultez la section Mettre à jour l'Agent SSM à l'aide de Fonctionnalité Exécuter la commande (p. 948).

Le tableau suivant répertorie les différences entre les versions majeures du schéma.

Version 1.2	Version 2.2 (dernière version)	Details (Détails)
runtimeConfig	mainSteps	Dans la version 2.2, la `mainSteps` section remplace `runtimeConfig` . Cette `mainSteps` section permet Systems Manager à d'exécuter les étapes dans l'ordre.
propriétés	entrées	Dans la version 2.2, la section `inputs` remplace la section `properties` . La section

Version 1.2	Version 2.2 (dernière version)	Details (Détails)
		`inputs` accepte des paramètres pour les étapes.
commands	runCommand	Dans la version 2.2, la section `inputs` prend le paramètre `runCommand` au lieu du paramètre `commands`.
id	action	Dans la version 2.2, `Action` remplace `ID` . Il s'agit simplement d'un changement de nom.
ne s'applique pas	nom	Dans la version 2.2, `name` est tout nom défini par l'utilisateur pour une étape.

Utilisation du paramètre de condition préalable

Avec la version de schéma 2.2 ou une version ultérieure, vous pouvez utiliser le paramètre `precondition` pour spécifier le système d'exploitation cible pour chaque plug-in. Le paramètre `precondition` prend en charge `platformType` et une valeur soit de `Windows`, soit de `Linux`.

Pour les documents utilisant la version de schéma 2.2 ou une version ultérieure, si `precondition` n'est pas spécifié, chaque plug-in est soit exécuté, soit ignoré en fonction de sa compatibilité avec le système d'exploitation. Pour les documents utilisant le schéma 2.0 ou antérieur, les plug-ins incompatibles entraînent une erreur.

Par exemple, dans un document de version de schéma 2.2, si `precondition` n'est pas spécifié et que le plug-in `aws:runShellScript` figure dans la liste, l'étape s'exécute sur les instances Linux, mais le système l'ignore sur les instances Windows Server, car le `aws:runShellScript` n'est pas compatible avec les instances Windows Server. Néanmoins, pour un document de version de schéma 2.0., si vous spécifiez le plug-in `aws:runShellScript`, puis exécutez le document sur des instances Windows Server, l'exécution échoue. Un exemple du paramètre de condition préalable dans un document SSM est fourni plus loin dans cette section.

Version de schéma 2.2

Éléments de niveau supérieur

L'exemple suivant présente les éléments supérieurs d'un document SSM qui utilise la version de schéma 2.2.

YAML

```
---
schemaVersion: "2.2"
description: A description of the document.
parameters:
  parameter 1:
    property 1: "value"
    property 2: "value"
  parameter 2:
    property 1: "value"
    property 2: "value"
mainSteps:
  - action: Plugin name
    name: A name for the step.
```

```
    inputs:
      input 1: "value"
      input 2: "value"
      input 3: "{{ parameter 1 }}"
```

JSON

```
{
   "schemaVersion": "2.2",
   "description": "A description of the document.",
   "parameters": {
       "parameter 1": {
           "property 1": "value",
           "property 2": "value"
        },
        "parameter 2":{
           "property 1": "value",
           "property 2": "value"
        }
    },
   "mainSteps": [
      {
         "action": "Plugin name",
         "name": "A name for the step.",
         "inputs": {
            "input 1": "value",
            "input 2": "value",
            "input 3": "{{ parameter 1 }}"
         }
      }
   ]
}
```

Exemple de version de schéma 2.2

L'exemple suivant utilise le `aws:runPowerShellScript` plug-in pour exécuter une PowerShell commande sur les instances cibles.

YAML

```
---
schemaVersion: "2.2"
description: "Example document"
parameters:
  Message:
    type: "String"
    description: "Example parameter"
    default: "Hello World"
mainSteps:
  - action: "aws:runPowerShellScript"
    name: "example"
    inputs:
      runCommand:
      - "Write-Output {{Message}}"
```

JSON

```
{
   "schemaVersion": "2.2",
   "description": "Example document",
   "parameters": {
```

```
      "Message": {
         "type": "String",
         "description": "Example parameter",
         "default": "Hello World"
      }
   },
   "mainSteps": [
      {
         "action": "aws:runPowerShellScript",
         "name": "example",
         "inputs": {
            "runCommand": [
               "Write-Output {{Message}}"
            ]
         }
      }
   ]
}
```

Exemple de paramètre de condition préalable dans la version de schéma 2.2

La version de schéma 2.2 fournit le support multiplateforme. Cela signifie que dans un même document SSM, vous pouvez spécifier un système d'exploitation différent pour différents plug-ins. Le support multiplateforme utilise le même paramètre `precondition` dans une étape, tel que décrit dans l'exemple suivant.

YAML

```
---
schemaVersion: '2.2'
description: cross-platform sample
mainSteps:
- action: aws:runPowerShellScript
  name: PatchWindows
  precondition:
    StringEquals:
    - platformType
    - Windows
  inputs:
    runCommand:
    - cmds
- action: aws:runShellScript
  name: PatchLinux
  precondition:
    StringEquals:
    - platformType
    - Linux
  inputs:
    runCommand:
    - cmds
```

JSON

```
{
   "schemaVersion": "2.2",
   "description": "cross-platform sample",
   "mainSteps": [
      {
         "action": "aws:runPowerShellScript",
         "name": "PatchWindows",
         "precondition": {
            "StringEquals": [
```

```
                "platformType",
                "Windows"
            ]
         },
         "inputs": {
            "runCommand": [
                "cmds"
            ]
         }
      },
      {
         "action": "aws:runShellScript",
         "name": "PatchLinux",
         "precondition": {
            "StringEquals": [
                "platformType",
                "Linux"
            ]
         },
         "inputs": {
            "runCommand": [
                "cmds"
            ]
         }
      }
   ]
}
```

Exemple de gestionnaire d'état de version de schéma 2.2

Vous pouvez utiliser le SSM document suivant avec State Manager pour télécharger et installer le logiciel ClamAV antivirus. State Manager applique une configuration spécifique, ce qui signifie que chaque fois que l'State Managerassociation est exécutée, le système vérifie si le ClamAV logiciel est installé. Si tel n'est pas le cas, State Manager réexécute ce document.

YAML

```
---
schemaVersion: '2.2'
description: State Manager Bootstrap Example
parameters: {}
mainSteps:
- action: aws:runShellScript
  name: configureServer
  inputs:
    runCommand:
    - sudo yum install -y httpd24
    - sudo yum --enablerepo=epel install -y clamav
```

JSON

```
{
   "schemaVersion": "2.2",
   "description": "State Manager Bootstrap Example",
   "parameters": {},
   "mainSteps": [
      {
         "action": "aws:runShellScript",
         "name": "configureServer",
         "inputs": {
            "runCommand": [
                "sudo yum install -y httpd24",
```

```
                "sudo yum --enablerepo=epel install -y clamav"
            ]
        }
    }
  ]
}
```

Exemple d'inventaire de version de schéma 2.2

Vous pouvez utiliser le document SSM suivant avec State Manager pour collecter les métadonnées d'inventaire relatives à vos instances.

YAML

```
---
schemaVersion: '2.2'
description: Software Inventory Policy Document.
parameters:
  applications:
    type: String
    default: Enabled
    description: "(Optional) Collect data for installed applications."
    allowedValues:
    - Enabled
    - Disabled
  awsComponents:
    type: String
    default: Enabled
    description: "(Optional) Collect data for AWS Components like amazon-ssm-agent."
    allowedValues:
    - Enabled
    - Disabled
  networkConfig:
    type: String
    default: Enabled
    description: "(Optional) Collect data for Network configurations."
    allowedValues:
    - Enabled
    - Disabled
  windowsUpdates:
    type: String
    default: Enabled
    description: "(Optional) Collect data for all Windows Updates."
    allowedValues:
    - Enabled
    - Disabled
  instanceDetailedInformation:
    type: String
    default: Enabled
    description: "(Optional) Collect additional information about the instance,
 including
      the CPU model, speed, and the number of cores, to name a few."
    allowedValues:
    - Enabled
    - Disabled
  customInventory:
    type: String
    default: Enabled
    description: "(Optional) Collect data for custom inventory."
    allowedValues:
    - Enabled
    - Disabled
mainSteps:
```

```
- action: aws:softwareInventory
  name: collectSoftwareInventoryItems
  inputs:
    applications: "{{ applications }}"
    awsComponents: "{{ awsComponents }}"
    networkConfig: "{{ networkConfig }}"
    windowsUpdates: "{{ windowsUpdates }}"
    instanceDetailedInformation: "{{ instanceDetailedInformation }}"
    customInventory: "{{ customInventory }}"
```

JSON

```
{
   "schemaVersion": "2.2",
   "description": "Software Inventory Policy Document.",
   "parameters": {
      "applications": {
         "type": "String",
         "default": "Enabled",
         "description": "(Optional) Collect data for installed applications.",
         "allowedValues": [
            "Enabled",
            "Disabled"
         ]
      },
      "awsComponents": {
         "type": "String",
         "default": "Enabled",
         "description": "(Optional) Collect data for AWS Components like amazon-ssm-
agent.",
         "allowedValues": [
            "Enabled",
            "Disabled"
         ]
      },
      "networkConfig": {
         "type": "String",
         "default": "Enabled",
         "description": "(Optional) Collect data for Network configurations.",
         "allowedValues": [
            "Enabled",
            "Disabled"
         ]
      },
      "windowsUpdates": {
         "type": "String",
         "default": "Enabled",
         "description": "(Optional) Collect data for all Windows Updates.",
         "allowedValues": [
            "Enabled",
            "Disabled"
         ]
      },
      "instanceDetailedInformation": {
         "type": "String",
         "default": "Enabled",
         "description": "(Optional) Collect additional information about the instance,
 including\nthe CPU model, speed, and the number of cores, to name a few.",
         "allowedValues": [
            "Enabled",
            "Disabled"
         ]
      },
      "customInventory": {
         "type": "String",
```

```
         "default": "Enabled",
         "description": "(Optional) Collect data for custom inventory.",
         "allowedValues": [
            "Enabled",
            "Disabled"
         ]
      }
   },
   "mainSteps": [
      {
         "action": "aws:softwareInventory",
         "name": "collectSoftwareInventoryItems",
         "inputs": {
            "applications": "{{ applications }}",
            "awsComponents": "{{ awsComponents }}",
            "networkConfig": "{{ networkConfig }}",
            "windowsUpdates": "{{ windowsUpdates }}",
            "instanceDetailedInformation": "{{ instanceDetailedInformation }}",
            "customInventory": "{{ customInventory }}"
         }
      }
   ]
}
```

Exemple AWS-ConfigureAWSPackage de version de schéma 2.2

L'exemple suivant présente le document AWS-ConfigureAWSPackage. La mainSteps section inclut le plug-in aws:configurePackage dans l'étape de l'action.

> **Note**
>
> Sur les systèmes d'exploitation Linux, seuls les packages `AmazonCloudWatchAgent` et `AWSSupport-EC2Rescue` sont pris en charge.

YAML

```
---
schemaVersion: '2.2'
description: 'Install or uninstall the latest version or specified version of an AWS
  package. Available packages include the following: AWSPVDriver, AwsEnaNetworkDriver,
  AwsVssComponents, and AmazonCloudWatchAgent, and AWSSupport-EC2Rescue.'
parameters:
  action:
    description: "(Required) Specify whether or not to install or uninstall the
 package."
    type: String
    allowedValues:
    - Install
    - Uninstall
  name:
    description: "(Required) The package to install/uninstall."
    type: String
    allowedPattern: "^arn:[a-z0-9][-.a-z0-9]{0,62}:[a-z0-9][-.a-z0-9]{0,62}:([a-
z0-9][-.a-z0-9]{0,62})?:([a-z0-9][-.a-z0-9]{0,62})?:package\\/[a-zA-Z][a-zA-Z0-9\\-_]
{0,39}$|^[a-zA-Z][a-zA-Z0-9\\-_]{0,39}$"
  version:
    type: String
    description: "(Optional) A specific version of the package to install or uninstall.
      If installing, the system installs the latest published version, by default.
      If uninstalling, the system uninstalls the currently installed version, by
 default.
      If no installed version is found, the latest published version is downloaded,
      and the uninstall action is run."
```

```
    default: latest
mainSteps:
- action: aws:configurePackage
  name: configurePackage
  inputs:
    name: "{{ name }}"
    action: "{{ action }}"
    version: "{{ version }}"
```

JSON

```
{
   "schemaVersion": "2.2",
   "description": "Install or uninstall the latest version or specified version
 of an AWS package. Available packages include the following: AWSPVDriver,
 AwsEnaNetworkDriver, AwsVssComponents, and AmazonCloudWatchAgent, and AWSSupport-
EC2Rescue.",
   "parameters": {
      "action": {
         "description":"(Required) Specify whether or not to install or uninstall the
 package.",
         "type":"String",
         "allowedValues":[
            "Install",
            "Uninstall"
         ]
      },
      "name": {
         "description": "(Required) The package to install/uninstall.",
         "type": "String",
         "allowedPattern": "^arn:[a-z0-9][-.a-z0-9]{0,62}:[a-z0-9][-.a-z0-9]{0,62}:([a-
z0-9][-.a-z0-9]{0,62})?:([a-z0-9][-.a-z0-9]{0,62})?:package\\/[a-zA-Z][a-zA-Z0-9\\-_]
{0,39}$|^[a-zA-Z][a-zA-Z0-9\\-_]{0,39}$"
      },
      "version": {
         "type": "String",
         "description": "(Optional) A specific version of the package to install
 or uninstall. If installing, the system installs the latest published version, by
 default. If uninstalling, the system uninstalls the currently installed version, by
 default. If no installed version is found, the latest published version is downloaded,
 and the uninstall action is run.",
         "default": "latest"
      }
   },
   "mainSteps":[
      {
         "action": "aws:configurePackage",
         "name": "configurePackage",
         "inputs": {
            "name": "{{ name }}",
            "action": "{{ action }}",
            "version": "{{ version }}"
         }
      }
   ]
}
```

Version de schéma 1.2

L'exemple suivant présente les éléments supérieurs d'un document de version de schéma 1.2.

```
{
```

```
   "schemaVersion":"1.2",
   "description":"A description of the SSM document.",
   "parameters":{
      "parameter 1":{
         "one or more parameter properties"
      },
      "parameter 2":{
         "one or more parameter properties"
      },
      "parameter 3":{
         "one or more parameter properties"
      }
   },
   "runtimeConfig":{
      "plugin 1":{
         "properties":[
            {
               "one or more plugin properties"
            }
         ]
      }
   }
}
```

Exemple aws:runShellScript de version de schéma 1.2

L'exemple suivant présente le document SSM AWS-RunShellScript. La runtimeConfig section inclut le plug-in aws:runShellScript.

```
{
    "schemaVersion":"1.2",
    "description":"Run a shell script or specify the commands to run.",
    "parameters":{
        "commands":{
            "type":"StringList",
            "description":"(Required) Specify a shell script or a command to run.",
            "minItems":1,
            "displayType":"textarea"
        },
        "workingDirectory":{
            "type":"String",
            "default":"",
            "description":"(Optional) The path to the working directory on your instance.",
            "maxChars":4096
        },
        "executionTimeout":{
            "type":"String",
            "default":"3600",
            "description":"(Optional) The time in seconds for a command to complete before
 it is considered to have failed. Default is 3600 (1 hour). Maximum is 172800 (48 hours).",
            "allowedPattern":"([1-9][0-9]{0,3})|(1[0-9]{1,4})|(2[0-7][0-9]{1,3})|(28[0-7]
[0-9]{1,2})|(28800)"
        }
    },
    "runtimeConfig":{
        "aws:runShellScript":{
            "properties":[
                {
                    "id":"0.aws:runShellScript",
                    "runCommand":"{{ commands }}",
                    "workingDirectory":"{{ workingDirectory }}",
                    "timeoutSeconds":"{{ executionTimeout }}"
                }
            ]
```

```
        }
    }
}
```

Version de schéma 0.3

Éléments de niveau supérieur

L'exemple suivant présente les éléments supérieurs d'un document de version de schéma 0.3 ou ultérieur au format JSON.

```
{
    "description": "document-description",
    "schemaVersion": "0.3",
    "assumeRole": "{{assumeRole}}",
    "parameters": {
        "parameter-1": {
            "type": "String",
            "description": "parameter-1-description",
            "default": ""
        },
        {
        "parameter-2": {
            "type": "String",
            "description": "parameter-2-description",
            "default": ""
        }
    },
    "mainSteps": [
        {
            "name": "myStepName",
            "action": "action-name",
            "maxAttempts": 1,
            "inputs": {
                "Handler": "python-only-handler-name",
                "Runtime": "runtime-name",
                "Attachment": "script-or-zip-name"
            },
            "outputs": {
                "Name": "output-name",
                "Selector": "selector.value",
                "Type": "data-type"
            }
        }
    ],
    "files": {
        "script-or-zip-name": {
            "checksums": {
                "sha256": "checksum"
            },
            "size": 1234
        }
    }
}
```

Exemple de document d'automatisation YAML

L'exemple suivant montre le contenu d'un document Automation, au format YAML. Cet exemple fonctionnel de la version 0.3 du schéma de document illustre également l'utilisation de Markdown pour formater les descriptions de documents.

```
description: >-
```

```
  ##Title: LaunchInstanceAndCheckState

  -----

  **Purpose**: This Automation document first launches an EC2 instance
  using the AMI ID provided in the parameter ```imageId```. The second step of
  this document continuously checks the instance status check value for the
  launched instance until the status ```ok``` is returned.


  ##Parameters:

  -----

  Name | Type | Description | Default Value

  ------------- | ------------- | ------------- | -------------

  assumeRole | String | (Optional) The ARN of the role that allows Automation to
  perform the actions on your behalf. | -

  imageId  | String | (Optional) The AMI ID to use for launching the instance.
  The default value uses the latest Amazon Linux AMI ID available. | {{
  ssm:/aws/service/ami-amazon-linux-latest/amzn-ami-hvm-x86_64-gp2 }}
schemaVersion: '0.3'
assumeRole: 'arn:aws:iam::111122223333::role/AutomationServiceRole'
parameters:
  imageId:
    type: String
    default: '{{ ssm:/aws/service/ami-amazon-linux-latest/amzn-ami-hvm-x86_64-gp2 }}'
    description: >-
      (Optional) The AMI ID to use for launching the instance. The default value
      uses the latest released Amazon Linux AMI ID.
  tagValue:
    type: String
    default: ' LaunchedBySsmAutomation'
    description: >-
      (Optional) The tag value to add to the instance. The default value is
      LaunchedBySsmAutomation.
  instanceType:
    type: String
    default: t2.micro
    description: >-
      (Optional) The instance type to use for the instance. The default value is
      t2.micro.
mainSteps:
  - name: LaunchEc2Instance
    action: 'aws:executeScript'
    outputs:
      - Name: payload
        Selector: $.Payload
        Type: StringMap
    inputs:
      Runtime: python3.6
      Handler: launch_instance
      Script: ''
      InputPayload:
        image_id: '{{ imageId }}'
        tag_value: '{{ tagValue }}'
        instance_type: '{{ instanceType }}'
      Attachment: launch.py
    description: >-
      **About This Step**


      This step first launches an EC2 instance using the ```aws:executeScript```
```

```
      action and the provided python script.
  - name: WaitForInstanceStatusOk
    action: 'aws:executeScript'
    inputs:
      Runtime: python3.6
      Handler: poll_instance
      Script: |-
        def poll_instance(events, context):
          import boto3
          import time

          ec2 = boto3.client('ec2')

          instance_id = events['InstanceId']

          print('[INFO] Waiting for instance status check to report ok', instance_id)

          instance_status = "null"

          while True:
            res = ec2.describe_instance_status(InstanceIds=[instance_id])

            if len(res['InstanceStatuses']) == 0:
              print("Instance status information is not available yet")
              time.sleep(5)
              continue

            instance_status = res['InstanceStatuses'][0]['InstanceStatus']['Status']

            print('[INFO] Polling to get status of the instance', instance_status)

            if instance_status == 'ok':
              break

            time.sleep(10)

          return {'Status': instance_status, 'InstanceId': instance_id}
      InputPayload: '{{ LaunchEc2Instance.payload }}'
    description: >-
      **About This Step**


      The python script continuously polls the instance status check value for
      the instance launched in Step 1 until the ```ok``` status is returned.
files:
  launch.py:
    checksums:
      sha256: 18871b1311b295c43d0f...[truncated]...772da97b67e99d84d342ef4aEXAMPLE
```

Syntaxe du document SSM

La syntaxe de votre document est définie par la version de schéma utilisée lors de sa création. Nous vous recommandons d'utiliser la version de schéma 2.2 ou ultérieure pour les documents de commande. Les documents d'automatisation utilisent la version de schéma 0.3. De plus, les documents Automation prennent en charge l'utilisation de Markdown, un langage de balisage, qui vous permet d'ajouter des descriptions de style wiki aux documents et des étapes individuelles au sein du document. Pour plus d'informations sur l'utilisation de Markdown, consultez Utilisation de Markdown dans AWS.

Les éléments de niveau supérieur constituent la structure du document SSM. Les informations présentées dans cette rubrique concernent les documents SSM `Command` et `Automation`.

Éléments de niveau supérieur

schemaVersion

Version de schéma à utiliser.

Type: Version

Obligatoire: Oui

description

Informations que vous fournissez pour décrire l'objectif du document.

Type: String (chaîne)

Obligatoire: Non

paramètres

Structure qui définit les paramètres acceptés par le document. Pour les paramètres que vous référencez souvent, nous vous recommandons de les stocker dans Systems Manager Parameter Store, puis de les référencer. Vous pouvez référencer les paramètres Parameter Store `String` et `StringList` dans cette section d'un document. Vous pouvez référencer les paramètres Parameter Store `SecureString` dans cette section d'un document. Vous pouvez référencer un Parameter Store paramètre à l'aide du format suivant :

```
{{ssm:parameter-name}}
```

Pour plus d'informations sur Parameter Store, consultez AWS Systems Manager Parameter Store (p. 218).

Type: Structure

La structure `parameters` accepte les champs et valeurs suivants :

- `type`: (Obligatoire) Les valeurs autorisées sont notamment les suivantes : `String`, `StringList` , `Boolean` , et `MapList` .`StringMap` Pour voir des exemples de chaque type, consultez la section suivante.Exemples de `type` paramètres de document SSM (p. 1229)

  > **Note**
  >
  > Pour utiliser un nombre comme valeur de paramètre, utilisez `String` comme type de paramètre.

- `description`: (Facultatif) Description du paramètre.
- `default`: (Facultatif) Valeur par défaut du paramètre ou référence à un paramètre dans Parameter Store .
- `allowedValues`: (Facultatif) Tableau de valeurs autorisées pour le paramètre. La définition des valeurs autorisées pour le paramètre valide l'entrée utilisateur. Si un utilisateur saisit une valeur non autorisée, l'exécution échoue.

  YAML

  ```
  DirectoryType:
    type: String
    description: "(Required) The directory type to launch."
    default: AwsMad
    allowedValues:
    - AdConnector
    - AwsMad
    - SimpleAd
  ```

JSON

```
"DirectoryType": {
  "type": "String",
  "description": "(Required) The directory type to launch.",
  "default": "AwsMad",
  "allowedValues": [
    "AdConnector",
    "AwsMad",
    "SimpleAd"
  ]
}
```

- `allowedPattern`: (Facultatif) Expression régulière qui valide si l'entrée utilisateur correspond au modèle défini pour le paramètre. Si l'entrée utilisateur ne correspond pas au modèle autorisé, l'exécution échoue.

  YAML

```
InstanceId:
  type: String
  description: "(Required) The instance ID to target."
  allowedPattern: "^i-[a-z0-9]{8,17}$"
  default: ''
```

  JSON

```
"InstanceId": {
  "type": "String",
  "description": "(Required) The instance ID to target.",
  "allowedPattern": "^i-[a-z0-9]{8,17}$",
  "default": ""
}
```

- `displayType`: (Facultatif) Utilisé pour afficher un `textfield` ou un `textarea` dans l' AWS Management Console . `textfield`est une zone de texte à une seule ligne. `textarea`est une zone de texte à plusieurs lignes.
- `minItems`: (Facultatif) Nombre minimum d'éléments autorisés.
- `maxItems`: (Facultatif) Nombre maximal d'éléments autorisés.
- `minChars`: (Facultatif) Nombre minimal de caractères de paramètre autorisés.
- `maxChars`: (Facultatif) Nombre maximal de caractères de paramètre autorisés.

Obligatoire: Non

runtimeConfig

(Version de schéma 1.2 seulement) Configuration de l'instance telle qu'appliquée par un ou plusieurs plug-ins Systems Manager. L'exécution en séquence des plug-ins n'est pas garantie.

Type: Dictionnaire<string,PluginConfiguration>

Obligatoire: Non

mainSteps

(Version de schéma 0.3, 2.0 et 2.2 uniquement) Objet pouvant inclure plusieurs étapes (plug-ins). Les plug-ins sont définis en étapes. Les étapes s'exécutent par ordre séquentiel telles que listées dans le document.

Type: Dictionnaire<string,PluginConfiguration>

Obligatoire: Oui

outputs

(Version de schéma 0.3 uniquement) Données générées par l'exécution de ce document pouvant être utilisées dans d'autres processus. Par exemple, si votre document crée une nouvelle AMI, vous pouvez spécifier "CreateImage.ImageId" comme valeur de sortie, puis utiliser cette sortie pour créer de nouvelles instances lors d'une exécution d'automatisation ultérieure. Pour plus d'informations sur les sorties, consultez Utilisation des entrées et des sorties (p. 473).

Type: Dictionnaire<string,OutputConfiguration>

Obligatoire: Non

fichiers

(Version de schéma 0.3 uniquement) Les fichiers de script (et leurs sommes de contrôle) attachés au document et exécutés lors d'une exécution Automation. S'applique uniquement aux documents qui incluent l'action `aws:executeScript` et pour lesquels des pièces jointes ont été spécifiées dans une ou plusieurs étapes.

Pour la prise en charge de l'exécution des scripts, les documents Automation prennent actuellement en charge les scripts pour Python 3.6, Python 3.7 et PowerShell Core 6.0. Pour de plus amples informations sur l'inclusion de scripts dans les documents Automation, veuillez consulter Création de documents Automation qui exécutent des scripts (p. 451) et Procédure : Utilisation de Document Builder pour créer un document Automation personnalisé (p. 670).

Lorsque vous créez un document Automation ou un manuel stratégique, vous spécifiez les fichiers d'attachements à l'aide de l'option `--attachments` (pour l'AWS CLI) ou `Attachments` (pour l'API et le SDK). Vous pouvez spécifier l'emplacement du fichier pour les fichiers locaux et les fichiers stockés dans des compartiments Amazon S3.

YAML

```
---
files:
  launch.py:
    checksums:
      sha256: 18871b1311b295c43d0f...[truncated]...772da97b67e99d84d342ef4aEXAMPLE
```

JSON

```
"files": {
    "launch.py": {
        "checksums": {
            "sha256": "18871b1311b295c43d0f...
[truncated]...772da97b67e99d84d342ef4aEXAMPLE"
        }
    }
}
```

Type: Dictionnaire<string,FilesConfiguration>

Obligatoire: Non

Exemples de `type` paramètres de document SSM

Les types de paramètres des documents SSM sont statiques. Cela signifie que le type de paramètre ne peut pas être modifié après avoir été défini. Lorsque vous utilisez des paramètres avec des plugins de document SSM, le type d'un paramètre ne peut pas être modifié dynamiquement dans l'entrée d'un

plugin. Par exemple, vous ne pouvez pas référencer un paramètre `Integer` dans l'entrée `runCommand` du plugin `aws:runShellScript`, car cette entrée accepte une chaîne ou une liste de chaînes. Pour utiliser un paramètre pour une entrée de plugin, le type de paramètre doit correspondre au type accepté. Par exemple, vous devez spécifier un type de paramètre `Boolean` pour l'entrée `allowDowngrade` du plugin `aws:updateSsmAgent`. Si votre type de paramètre ne correspond pas au type d'entrée d'un plugin, le document SSM n'est pas validé et le système ne crée pas le document.

Lorsque vous utilisez des paramètres avec des actions SSM Automation, dans la plupart des cas les types de paramètres ne sont pas validés lorsque vous créez le document SSM. Ce n'est que lorsque vous utilisez l'action `aws:runCommand` que les types de paramètres sont validés lors de la création du document SSM. Dans tous les autres cas, la validation des paramètres se produit pendant l'exécution d'automatisation lorsque l'entrée d'une action est vérifiée avant d'exécuter cette dernière. Par exemple, si votre paramètre d'entrée est une `String` et que vous le référencez comme valeur pour l'entrée `MaxInstanceCount` de l'action `aws:runInstances`, le document SSM est créé. Toutefois, lors de l'exécution du document, l'automatisation échoue lors de la validation de l'action `aws:runInstances`, car l'entrée `MaxInstanceCount` nécessite un `Integer`.

Voici des exemples de chaque `type` de paramètre.

String (chaîne)

Une séquence de zéro ou plusieurs caractères Unicode entre guillemets. Par exemple, « i-1234567890abcdef0". Utilisez des barres obliques inverses comme caractères d'échappement.

YAML

```
---
InstanceId:
  type: String
  description: "(Required) The target EC2 instance ID."
```

JSON

```
"InstanceId":{
  "type":"String",
  "description":"(Required) The target EC2 instance ID."
}
```

StringList

Liste d'éléments String séparés par des virgules. Par exemple, ["cd ~", "pwd"].

YAML

```
---
commands:
  type: StringList
  description: "(Required) Specify a shell script or a command to run."
  minItems: 1
  displayType: textarea
```

JSON

```
"commands":{
  "type":"StringList",
  "description":"(Required) Specify a shell script or a command to run.",
  "minItems":1,
  "displayType":"textarea"
}
```

Booléen

Accepte uniquement `true` ou `false` . N'accepte pas la valeur « true » ou 0.

YAML

```
---
canRun:
  type: Boolean
  description: ''
  default: true
```

JSON

```
"canRun": {
  "type": "Boolean",
  "description": "",
  "default": true
}
```

Integer

Nombres entiers. N'accepte pas de nombres décimaux, par exemple 3,14159, ni de nombres entre guillemets, par exemple « 3 ».

YAML

```
---
timeout:
  type: Integer
  description: The type of action to perform.
  default: 100
```

JSON

```
"timeout": {
  "type": "Integer",
  "description": "The type of action to perform.",
  "default": 100
}
```

StringMap

Mappage de clés à des valeurs. Une clé ne peut être qu'une chaîne. Par exemple, {"Env": « Production »}.

YAML

```
---
notificationConfig:
  type: StringMap
  description: The configuration for events to be notified about
  default:
    NotificationType: Command
    NotificationEvents:
    - Failed
    NotificationArn: "$dependency.topicArn"
  maxChars: 150
```

JSON

```
"notificationConfig" : {
```

```
  "type" : "StringMap",
  "description" : "The configuration for events to be notified about",
  "default" : {
    "NotificationType" : "Command",
    "NotificationEvents" : ["Failed"],
    "NotificationArn" : "$dependency.topicArn"
  },
  "maxChars" : 150
}
```

MapList

Liste d'StringMapéléments.

YAML

```
blockDeviceMappings:
  type: MapList
  description: The mappings for the create image inputs
  default:
  - DeviceName: "/dev/sda1"
    Ebs:
      VolumeSize: '50'
  - DeviceName: "/dev/sdm"
    Ebs:
      VolumeSize: '100'
  maxItems: 2
```

JSON

```
"blockDeviceMappings":{
  "type":"MapList",
  "description":"The mappings for the create image inputs",
  "default":[
    {
      "DeviceName":"/dev/sda1",
      "Ebs":{
        "VolumeSize":"50"
      }
    },
    {
      "DeviceName":"/dev/sdm",
      "Ebs":{
        "VolumeSize":"100"
      }
    }
  ],
  "maxItems":2
}
```

Référence du plug-in de document Systems Manager

Cette référence décrit les plug-ins que vous pouvez spécifier dans un document de commande AWS Systems Manager (SSM). Ces plug-ins ne peuvent pas être utilisés dans les documents Automation SSM qui utilisent des actions Automation. Pour de plus amples informations sur les actions AWS Systems Manager Automation, veuillez consulter Référence sur les actions Systems Manager Automation (p. 386).

Systems Manager détermine les actions à effectuer sur une instance gérée en lisant le contenu d'un document SSM. Chaque document comprend une section d'exécution de code. En fonction de la version de schéma de votre document, cette section d'exécution de code peut inclure un ou plusieurs plug-ins,

ou bien une ou plusieurs étapes. Dans le cadre de cette rubrique d'aide, les plug-ins et les étapes sont appelés plug-ins. Cette section comprend des informations sur chacun des plug-ins Systems Manager. Pour plus d'informations sur les documents, la création de documents et les différences entre les versions de schéma, consultez Documents AWS Systems Manager (p. 1211).

> **Note**
>
> Certains plug-ins décrits ici s'exécutent uniquement sur des instances Windows Server ou sur des instances Linux. Les dépendances de plate-forme sont notées pour chaque plug-in.

Table des matières

- aws:applications (p. 1233)
- aws:cloudWatch (p. 1235)
- aws:configureDocker (p. 1242)
- aws:configurePackage (p. 1243)
- aws:domainJoin (p. 1245)
- aws:downloadContent (p. 1248)
- aws:psModule (p. 1250)
- aws:refreshAssociation (p. 1252)
- aws:runDockerAction (p. 1253)
- aws:runDocument (p. 1255)
- aws:runPowerShellScript (p. 1257)
- aws : runShellScript (p. 1259)
- aws:softwareInventory (p. 1261)
- aws:updateAgent (p. 1263)
- aws:updateSsmAgent (p. 1265)

aws:applications

Installer, réparer ou désinstaller des applications sur une instance EC2. Ce plug-in s'exécute uniquement sur les systèmes d'exploitation Windows Server. Pour plus d'informations, consultez la section Documents AWS Systems Manager (p. 1211).

Syntax

Schéma 2.2

YAML

```
---
schemaVersion: '2.2'
description: aws:applications plugin
parameters:
  source:
    description: "(Required) Source of msi."
    type: String
mainSteps:
- action: aws:applications
  name: example
  inputs:
    action: Install
    source: "{{ source }}"
```

JSON

```
{
  "schemaVersion":"2.2",
  "description":"aws:applications",
  "parameters":{
    "source":{
    "description":"(Required) Source of msi.",
    "type":"String"
    }
  },
  "mainSteps":[
    {
      "action":"aws:applications",
      "name":"example",
      "inputs":{
        "action":"Install",
        "source":"{{ source }}"
      }
    }
  ]
}
```

Schéma 1.2

YAML

```
---
runtimeConfig:
  aws:applications:
    properties:
    - id: 0.aws:applications
      action: "{{ action }}"
      parameters: "{{ parameters }}"
      source: "{{ source }}"
      sourceHash: "{{ sourceHash }}"
```

JSON

```
{
   "runtimeConfig":{
      "aws:applications":{
         "properties":[
            {
               "id":"0.aws:applications",
               "action":"{{ action }}",
               "parameters":"{{ parameters }}",
               "source":"{{ source }}",
               "sourceHash":"{{ sourceHash }}"
            }
         ]
      }
   }
}
```

Properties

action

Action à effectuer.

Type: Enum

Valeurs valides : `Install| | |RepairUninstall`

Obligatoire: Oui

paramètres

Paramètres pour le programme d'installation.

Type: String (chaîne)

Obligatoire: Non

source

URL du fichier `.msi` pour l'application.

Type: String (chaîne)

Obligatoire: Oui

sourceHash

Hachage SHA256 du fichier `.msi`.

Type: String (chaîne)

Obligatoire: Non

aws:cloudWatch

Exportez les données de Windows Server vers Amazon CloudWatch ou Amazon CloudWatch Logs et contrôlez les données à l'aide des métriques CloudWatch. Ce plug-in s'exécute uniquement sur les systèmes d'exploitation Windows Server. Pour plus d'informations sur la configuration de l'intégration de CloudWatch à Amazon EC2, consultez Envoi de journaux, d'événements et de compteurs de performances à Amazon CloudWatch. Pour plus d'informations sur les documents, consultez Documents AWS Systems Manager (p. 1211).

Important

Ce plug-in est obsolète. L'agent Agent CloudWatch unifié a remplacé Agent SSM comme outil d'envoi des données de journaux à Amazon CloudWatch Logs. Nous vous recommandons d'utiliser l'agent Agent CloudWatch unifié pour vos processus de collecte de journal. Pour plus d'informations, consultez les rubriques suivantes :

- Envoi de journaux d'instance à CloudWatch Logs (Agent CloudWatch) (p. 1327)
- Migrer la collecte de journaux d'instances Windows Server vers l'Agent CloudWatch (p. 1328)
- Collecte des métriques depuis les instances Amazon Elastic Compute Cloud et les serveurs locaux avec l'Agent CloudWatch dans le Guide de l'utilisateur Amazon CloudWatch

Vous pouvez exporter et contrôler les types de données suivants :

ApplicationEventLog

Envoie des données de journal des événements d'application à CloudWatch Logs

CustomLogs

Envoie n'importe quel fichier journal à base de texte à Amazon CloudWatch Logs . Le plug-in CloudWatch crée une empreinte pour les fichiers journaux. Le système associe ensuite un décalage

des données à chaque empreinte. Le plug-in charge les fichiers en cas de modifications, enregistre le décalage et associe celui-ci à une empreinte. Cette méthode est utilisée pour éviter que le système charge tous le fichiers si un utilisateur active le plug-in et associe le service à un répertoire contenant un grand nombre de fichiers.

> **Warning**
>
> N'oubliez pas que si votre application tronque ou tente de nettoyer des journaux au cours d'une interrogation, tous les journaux spécifiés pour `LogDirectoryPath` peuvent perdre des entrées. Si, par exemple, vous souhaitez limiter la taille des fichiers journaux, créez un nouveau fichier lorsque cette limite est atteinte, puis continuez à écrire les données dans le nouveau fichier.

ETW

Envoie des données de suivi d'événements pour Windows (ETW) à CloudWatch Logs.

IIS

Envoie des données de journal IIS à CloudWatch Logs.

PerformanceCounter

Envoie des compteurs de performances Windows à CloudWatch . Vous pouvez sélectionner plusieurs catégories à télécharger vers CloudWatch en tant que métriques. Pour chaque compteur de performances que vous souhaitez charger, créez une PerformanceCounter section avec un ID unique (par exemple, « PerformanceCounter2 », « PerformanceCounter3 », etc.) et configurez ses propriétés.

> **Note**
>
> Si l'Agent SSMou le plug-in CloudWatch est arrêté, les données du compteur de performances ne sont pas consignées dans CloudWatch. Ce comportement diffère de celui des journaux personnalisés ou des journaux d'événements Windows. Les journaux personnalisés et les journaux d'événements Windows préservent les données du compteur de performances et les chargent dans CloudWatch lorsque Agent SSM ou le plug-in CloudWatch est disponible.

SecurityEventLog

Envoie des données de journal des événements de sécurité à CloudWatch Logs.

SystemEventLog

Envoie des données de journal des événements système à CloudWatch Logs.

Vous pouvez définir les destinations suivantes pour les données :

CloudWatch

La destination où vos données de métriques de compteur de performances sont envoyées. Vous pouvez ajouter d'autres sections avec un ID unique IDs (par exemple, « CloudWatch2 », CloudWatch3 « , etc.) et spécifier une région différente pour chaque nouvel ID afin d'envoyer les mêmes données à différents emplacements.

CloudWatchLogs

La destination où vos données de journaux sont envoyées. Vous pouvez ajouter d'autres sections avec un seul IDs (par exemple, « CloudWatchLogs2 », CloudWatchLogs3 « , etc.) et spécifier une région différente pour chaque nouvel ID afin d'envoyer les mêmes données à différents emplacements.

Syntax

```
"runtimeConfig":{
```

```
        "aws:cloudWatch":{
            "settings":{
                "startType":"{{ status }}"
            },
            "properties":"{{ properties }}"
        }
    }
```

Paramètres et propriétés

AccessKey

Votre ID de clé d'accès rapide . Cette propriété est obligatoire, sauf si vous avez lancé votre instance par l'intermédiaire d'une rôle IAM. Elle ne peut pas être utilisée avec SSM.

Type: String (chaîne)

Obligatoire: Non

CategoryName

Catégorie du compteur de performances en provenance de Performance Monitor.

Type: String (chaîne)

Obligatoire: Oui

CounterName

Nom du compteur de performances en provenance de Performance Monitor.

Type: String (chaîne)

Obligatoire: Oui

CultureName

Paramètres régionaux où l'horodatage est consigné. Si CultureName est vide, il prend par défaut les mêmes paramètres régionaux que ceux actuellement utilisés par votre Windows Server instance.

Type: String (chaîne)

Valeurs valides : Pour obtenir la liste des valeurs prises en charge, consultez NLS (National Language Support) sur le site web de Microsoft. Notez que les valeurs div, div-MV, hu et hu-HU ne sont pas prises en charge.

Obligatoire: Non

DimensionName

Dimension pour votre métrique Amazon CloudWatch. Si vous spécifiez `DimensionName` , vous devez spécifier `DimensionValue` . Ces paramètres fournissent une autre vue lors de la création de la liste des métriques. Vous pouvez utiliser la même dimension pour plusieurs métriques afin d'afficher toutes les métriques appartenant à une dimension spécifique.

Type: String (chaîne)

Obligatoire: Non

DimensionValue

Valeur de dimension pour votre métrique Amazon CloudWatch.

Type: String (chaîne)

Obligatoire: Non

Encodage

Encodage de fichier à utiliser (par exemple, UTF-8). Utilisez le nom d'encodage, pas le nom complet.

Type: String (chaîne)

Valeurs valides : Pour obtenir la liste des valeurs prises en charge, consultez Classe d'encodage dans la bibliothèque MSDN.

Obligatoire: Oui

Filtrer

Préfixe des noms de journaux. Laissez ce paramètre vide de façon à surveiller tous les fichiers.

Type: String (chaîne)

Valeurs valides : Pour obtenir la liste des valeurs prises en charge, consultez la FileSystemWatcherFilter propriété dans la bibliothèque MSDN.

Obligatoire: Non

Flux

Chaque type de données à télécharger, avec la destination des données (CloudWatch ou CloudWatch Logs). Par exemple, pour envoyer un compteur de performances défini dans `"Id": "PerformanceCounter"` à la destination CloudWatch définie dans `"Id": "CloudWatch"`, entrez "PerformanceCounter,CloudWatch". De même, pour envoyer le journal personnalisé, le journal ETW et le journal système à la destination CloudWatch Logs définie dans `"Id": "ETW"`, entrez "(ETW),CloudWatchLogs". En outre, vous pouvez envoyer le même compteur de performances ou fichier journal à plus d'une destination. Par exemple, pour envoyer le journal d'application à deux destinations différentes que vous avez définies sous `"Id": "CloudWatchLogs"` et `"Id": "CloudWatchLogs2"` , entrez « ApplicationEventLog,(CloudWatchLogs, CloudWatchLogs2) ».

Type: String (chaîne)

Valeurs valides (source) : `ApplicationEventLog`| | `CustomLogs` | `ETW` |`PerformanceCounterSystemEventLogSecurityEventLog`

Valeurs valides (destination) : `CloudWatch`| | |`CloudWatchLogsCloudWatch`*n* | `CloudWatchLogs`*n*

Obligatoire: Oui

FullName

Nom complet du composant.

Type: String (chaîne)

Obligatoire: Oui

Id

Identifie la source ou la destination des données. Cet identificateur doit être unique au sein du fichier de configuration.

Type: String (chaîne)

Obligatoire: Oui

InstanceName

Nom de l'instance du compteur de performances. N'utilisez pas d'astérisque (*) pour indiquer toutes les instances, car chaque composant du compteur de performance prend en charge un seul élément. Cependant, vous pouvez utiliser _Total.

Type: String (chaîne)

Obligatoire: Oui

Niveaux

Types des messages à envoyer à Amazon CloudWatch.

Type: String (chaîne)

Valeurs valides :

- 1 - Uniquement les messages d'erreur chargés.
- 2 - Uniquement les messages d'avertissement chargés.
- 4 - Uniquement les messages d'information chargés.

Notez que vous pouvez ajouter des valeurs pour inclure plusieurs types de message. Par exemple, 3 signifie que les messages d'erreur (1) et les messages d'avertissement (2) sont inclus. Une valeur de 7 signifie que les messages d'erreur (1), les messages d'avertissement (2) et les messages d'information (4) sont inclus.

Obligatoire: Oui

> **Note**
>
> Pour les journaux de sécurité Windows, Levels doit être défini sur 7.

LineCount

Nombre de lignes de l'en-tête permettant d'identifier le fichier journal. Par exemple, les fichiers journaux IIS ont des en-têtes presque identiques. Vous pouvez entrer 3, qui lirait les trois premières lignes de l'en-tête du fichier journal pour l'identifier. Dans les fichiers journaux IIS, la troisième ligne correspond à l'horodatage qui diffère d'un fichier journal à l'autre.

Type: Integer

Obligatoire: Non

LogDirectoryPath

Pour CustomLogs , chemin d'accès où les journaux sont stockés sur votre instance EC2. Pour les journaux IIS, dossier dans lequel les journaux IIS sont stockés pour un site individuel (par exemple, C:\\\\inetpub\\\\logs\\\\LogFiles\\\\W3SVC*n*. Pour les journaux IIS, seul le format de journal W3C est pris en charge. Les formats IIS, NCSA et Personnalisé ne sont pas pris en charge.

Type: String (chaîne)

Obligatoire: Oui

LogGroup

Nom de votre groupe de journaux. Ce nom s'affiche sur l'écran Groupes de journaux dans la console CloudWatch.

Type: String (chaîne)

Obligatoire: Oui

LogName

Nom du fichier journal.

1. Pour trouver le nom du journal, dans l'observateur d'événements, dans le panneau de navigation, cliquez sur Journaux d'applications et de services.
2. Dans la liste des journaux, cliquez avec le bouton droit de la souris sur le journal que vous souhaitez télécharger (par exemple, Microsoft > Windows > Sauvegarde > Opérationnelle), puis cliquez sur Créer une vue personnalisée.
3. Dans la boîte de dialogue Créer une vue personnalisée, cliquez sur l'onglet XML. LLogName' est dans la <Select Path=>balise (par exemple, `Microsoft-Windows-Backup`). Copiez ce texte dans le LogName paramètre.

Type: String (chaîne)

Valeurs valides : `Application| | |SecuritySystemMicrosoft-Windows-WinINet/Analytic`

Obligatoire: Oui

LogStream

Flux de journal de destination. Si vous utilisez {instance_id}, la valeur par défaut, l'ID d'instance de cette instance est utilisée en tant que nom du flux de journal.

Type: String (chaîne)

Valeurs valides : `{instance_id}| | |{hostname}{ip_address}` *`<log_stream_name>`*

Si vous entrez un nom de flux de journal qui n'existe pas déjà, CloudWatch Logs le crée automatiquement pour vous. Vous pouvez utiliser une chaîne littérale ou des variables prédéfinies ({instance_id}, {hostname}, {ip_address}), ou une combinaison des trois pour définir un nom de flux de journal.

Le nom de flux de journaux spécifié dans ce paramètre s'affiche dans Groupes de journaux > Flux pour ***`<YourLogStream>`***dans la console .CloudWatch

Obligatoire: Oui

MetricName

Métrique CloudWatch sous laquelle vous souhaitez que les données de performances apparaissent.

> **Note**
>
> N'utilisez pas de caractères spéciaux dans ce nom. Sinon, la métrique et les alarmes associées risquent de ne pas fonctionner.

Type: String (chaîne)

Obligatoire: Oui

NameSpace

Namespace de la métrique où vous voulez écrire les données des compteurs de performances.

Type: String (chaîne)

Obligatoire: Oui

PollInterval

Nombre de secondes avant qu'un nouveau compteur de performances et des données de journal soient chargés.

Type: Integer

Valeurs valides : Définissez cette valeur sur 5 secondes ou plus. La valeur recommandée est 15 secondes (00:00:15).

Obligatoire: Oui

Région

Région AWS où vous souhaitez envoyer les données de journaux. Même si vous pouvez envoyer des compteurs de performances à une autre région où vous envoyez vos données de journal, nous vous recommandons de définir ce paramètre sur la même région où votre instance est en cours d'exécution.

Type: String (chaîne)

Valeurs valides : Régions IDs des régions AWS prises en charge par Systems Manager et CloudWatch Logs , comme `us-east-2` , et `eu-west-1` .`ap-southeast-1` Pour obtenir la liste des régions AWS prises en charge par chaque service, consultez Points de terminaison de Amazon CloudWatch Logs service et Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services .

Obligatoire: Oui

SecretKey

Clé d'accès secrète de votre compte . Cette propriété est obligatoire, sauf si vous avez lancé votre instance par l'intermédiaire d'une rôle IAM.

Type: String (chaîne)

Obligatoire: Non

startType

Activer ou désactiver CloudWatch sur l'instance.

Type: String (chaîne)

Valeurs valides : `Enabled`|`Disabled`

Obligatoire: Oui

TimestampFormat

Format d'horodatage que vous voulez utiliser. Pour obtenir une liste des valeurs prises en charge, consultez Chaînes de format de date et d'heure personnalisées dans la bibliothèque MSDN.

Type: String (chaîne)

Obligatoire: Oui

TimeZoneKind

Fournit des informations de fuseau horaire si aucune n'est comprise dans l'horodatage de vos journaux. Si ce paramètre est laissé vide et si votre horodatage n'inclut pas les informations de fuseau horaire, CloudWatch Logs utilise le fuseau horaire local par défaut. Ce paramètre est ignoré si votre horodatage contient déjà des informations de fuseau horaire.

Type: String (chaîne)

Valeurs valides : `Local`|`UTC`

Obligatoire: Non

Unit

Unité de mesure appropriée pour la métrique.

Type: String (chaîne)

Valeurs valides : Secondes | Microsecondes | Millisecondes | Octets | Kilooctets | Megaoctets | Gigaoctets | Teraoctets | Teraoctets | Bits | Kilobits | Megabits | Gigabits | Terabits | Percent | Count | Bytes/Second | Megalobytes/Seconds | MegaSeconds | Gigabytes/Seconds | Terabytes | Terabits | Counts/Seconds (nombre de kilobits/kilobits/secondes/secondes/secondes/secondes/secondes | secondes |secondes | gigaoctets | gigaoctets | gigaoctets | gigaoctets | gigaoctets |s |s |s |s |s |s |s |s |s |s |s |s |s/secondes

Obligatoire: Oui

aws:configureDocker

(Version de schéma 2.0 ou ultérieure) Configuration d'une instance pour un fonctionnement avec des conteneurs et Docker. Ce plug-in est pris en charge sur les systèmes d'exploitation Linux et Windows Server. Pour plus d'informations, consultez la section Documents AWS Systems Manager (p. 1211).

Syntax

Schéma 2.2

YAML

```
---
schemaVersion: '2.2'
description: aws:configureDocker
parameters:
  action:
    description: "(Required) The type of action to perform."
    type: String
    default: Install
    allowedValues:
    - Install
    - Uninstall
mainSteps:
- action: aws:configureDocker
  name: configureDocker
  inputs:
    action: "{{ action }}"
```

JSON

```
{
  "schemaVersion": "2.2",
  "description": "aws:configureDocker plugin",
  "parameters": {
    "action": {
      "description": "(Required) The type of action to perform.",
      "type": "String",
      "default": "Install",
      "allowedValues": [
        "Install",
        "Uninstall"
      ]
    }
  },
```

```
  "mainSteps": [
    {
      "action": "aws:configureDocker",
      "name": "configureDocker",
      "inputs": {
        "action": "{{ action }}"
      }
    }
  ]
}
```

Inputs

action

Type d'action à effectuer.

Type: Enum

Valeurs valides : `Install`|`Uninstall`

Obligatoire: Oui

aws:configurePackage

(Version de schéma 2.0 ou ultérieure) Installez ou désinstallez un package Distributor. Vous pouvez installer la dernière version, la version par défaut ou une version du package que vous spécifiez. Les packages fournis par AWS sont également pris en charge. Ce plug-in s'exécute sur Windows Server et les systèmes d'exploitation Linux, mais tous les packages disponibles ne sont pas pris en charge sur les systèmes d'exploitation Linux.

Les packages AWS disponibles pour Windows Server sont les suivants : AWSPVDriver, AWSNVMe , AwsEnaNetworkDriver , AwsVssComponents , et AmazonCloudWatchAgent -EC2Rescue.AWSSupport

Les packages AWS disponibles pour les systèmes d'exploitation Linux sont les suivants : AmazonCloudWatchAgentet AWSSupport -EC2Rescue.

Pour plus d'informations, consultez la section Documents AWS Systems Manager (p. 1211).

Syntax

Schéma 2.2

YAML

```
---
schemaVersion: '2.2'
description: aws:configurePackage
parameters:
  name:
    description: "(Required) The name of the AWS package to install or uninstall."
    type: String
  action:
    description: "(Required) The type of action to perform."
    type: String
    default: Install
    allowedValues:
    - Install
    - Uninstall
```

```
mainSteps:
- action: aws:configurePackage
  name: configurePackage
  inputs:
    name: "{{ name }}"
    action: "{{ action }}"
    additionalArguments:
      SSM_parameter_store_arg: "{{ ssm:parameter_store_arg }}"
      SSM_custom_arg: "myValue"
```

JSON

```
{
  "schemaVersion": "2.2",
  "description": "aws:configurePackage",
  "parameters": {
    "name": {
      "description": "(Required) The name of the AWS package to install or uninstall.",
      "type": "String"
    },
    "action": {
      "description": "(Required) The type of action to perform.",
      "type": "String",
      "default": "Install",
      "allowedValues": [
        "Install",
        "Uninstall"
      ]
    }
  },
  "mainSteps": [
    {
      "action": "aws:configurePackage",
      "name": "configurePackage",
      "inputs": {
        "name": "{{ name }}",
        "action": "{{ action }}",
        "additionalArguments": {
          "SSM_parameter_store_arg": "{{ ssm:parameter_store_arg }}",
          "SSM_custom_arg": "myValue"
        }
      }
    }
  ]
}
```

Inputs

nom

Nom du package AWS à installer ou désinstaller. Les packages disponibles sont les suivants : AWSPVDriver, AwsEnaNetworkDriver , et AwsVssComponents .AmazonCloudWatchAgent

Type: String (chaîne)

Obligatoire: Oui

action

Installe ou désinstalle un package.

Type: Enum

Valeurs valides : `Install|Uninstall`

Obligatoire: Oui

installationType

Type d'installation à effectuer. Si vous spécifiez `Uninstall and reinstall`, le package est complètement désinstallé, puis réinstallé. L'application n'est pas disponible tant que la réinstallation n'est pas terminée. Si vous spécifiez`In-place update`, seuls les fichiers nouveaux ou modifiés sont ajoutés à l'installation existante, conformément aux instructions que vous fournissez dans un script mis à jour. L'application reste disponible tout au long du processus de mise à jour. L`In-place update`'option n'est pas prise en charge pour les packages publiés par AWS. `Uninstall and reinstall` est la valeur par défaut.

Type: Enum

Valeurs valides : `Uninstall and reinstall|In-place update`

Obligatoire: Non

additionalArguments

Paramètres supplémentaires à fournir à vos scripts d'installation, de désinstallation ou de mise à jour. Chaque paramètre doit être préfixé par `SSM_` . Vous pouvez référencer un Parameter Store paramètre dans vos arguments supplémentaires à l'aide de la convention `{{ssm:parameter-name}}` . Pour utiliser le paramètre supplémentaire dans votre script d'installation, de désinstallation ou de mise à jour, vous devez référencer le paramètre en tant que variable d'environnement à l'aide de la syntaxe appropriée pour le système d'exploitation. Par exemple, dans PowerShell , vous référencez l'`SSM_arg`argument comme `$Env:SSM_arg` suit : . Il n'y a pas de limite au nombre d'arguments que vous définissez, mais l'entrée d'argument supplémentaire a une limite de 4 096 caractères. Cette limite inclut toutes les clés et valeurs que vous définissez.

Type: StringMap

Obligatoire: Non

version

Version spécifique du package à installer ou désinstaller. Dans le cas d'une installation, le système installe par défaut la version publiée la plus récente. Dans le cas d'une désinstallation, le système désinstalle par défaut la version actuellement installée. Si aucune version installée n'est détectée, la version publiée la plus récente est téléchargée et l'action de désinstallation est exécutée.

Type: String (chaîne)

Obligatoire: Non

aws:domainJoin

Joindre une instance EC2 à un domaine. Ce plug-in s'exécute sur les systèmes d'exploitation Linux et Windows Server. Pour plus d'informations sur la jonction d'instances EC2, consultez Joindre une instance EC2 à votre annuaire AWS Managed Microsoft AD dans le AWS Directory Service Administration Guide . Pour plus d'informations sur les documents, consultez Documents AWS Systems Manager (p. 1211).

Syntax

Schéma 2.2

YAML

```
---
```

```
schemaVersion: '2.2'
description: aws:domainJoin
parameters:
  directoryId:
    description: "(Required) The ID of the directory."
    type: String
  directoryName:
    description: "(Required) The name of the domain."
    type: String
  dnsIpAddresses:
    description: "(Required) The IP addresses of the DNS servers for your directory."
    type: StringList
mainSteps:
- action: aws:domainJoin
  name: domainJoin
  inputs:
    directoryId: "{{ directoryId }}"
    directoryName: "{{ directoryName }}"
    directoryOU: "{{ directoryOU }}"
    dnsIpAddresses: "{{ dnsIpAddresses }}"
```

JSON

```
{
  "schemaVersion": "2.2",
  "description": "aws:domainJoin",
  "parameters": {
    "directoryId": {
      "description": "(Required) The ID of the directory.",
      "type": "String"
    },
    "directoryName": {
      "description": "(Required) The name of the domain.",
      "type": "String"
    },
    "dnsIpAddresses": {
      "description": "(Required) The IP addresses of the DNS servers for your
 directory.",
      "type": "StringList"
    },
  },
  "mainSteps": [
    {
      "action": "aws:domainJoin",
      "name": "domainJoin",
      "inputs": {
        "directoryId": "{{ directoryId }}",
        "directoryName": "{{ directoryName }}",
        "directoryOU":"{{ directoryOU }}",
        "dnsIpAddresses":"{{ dnsIpAddresses }}"
      }
    }
  ]
}
```

Schéma 1.2

YAML

```
---
runtimeConfig:
  aws:domainJoin:
```

```
    properties:
      directoryId: "{{ directoryId }}"
      directoryName: "{{ directoryName }}"
      directoryOU: "{{ directoryOU }}"
      dnsIpAddresses: "{{ dnsIpAddresses }}"
```

JSON

```
{
   "runtimeConfig":{
      "aws:domainJoin":{
         "properties":{
            "directoryId":"{{ directoryId }}",
            "directoryName":"{{ directoryName }}",
            "directoryOU":"{{ directoryOU }}",
            "dnsIpAddresses":"{{ dnsIpAddresses }}"
         }
      }
   }
}
```

Properties

directoryId

ID du répertoire.

Type: String (chaîne)

Obligatoire: Oui

Exemple : "directoryId": "d-1234567890"

directoryName

Le nom du domaine.

Type: String (chaîne)

Obligatoire: Oui

Exemple : "directoryName": "example.com"

directoryOU

Unité d'organisation (UO).

Type: String (chaîne)

Obligatoire: Non

Exemple : "directoryOU": "UO=test,DC=exemple,DC=com"

dnsIpAddresses

Adresses IP des serveurs DNS.

Type: StringList

Obligatoire: Oui

Exemple : « dnsIpAddresses » : [« 198.51.100.1 », 198.51.100.2"]

Examples

Pour obtenir des exemples, consultez Joindre une instance Windows Server à un domaine AWS Directory Service dans le Amazon EC2 Guide de l'utilisateur pour les instances Windows.

aws:downloadContent

(Version de schéma 2.0 ou ultérieure) Téléchargement de scripts et de documents SSM à partir d'emplacements distants. Ce plug-in est pris en charge sur les systèmes d'exploitation Linux et Windows Server.

Syntax

Schéma 2.2

YAML

```
---
schemaVersion: '2.2'
description: aws:downloadContent
parameters:
  sourceType:
    description: "(Required) The download source."
    type: String
  sourceInfo:
    description: "(Required) The information required to retrieve the content from
      the required source."
    type: StringMap
mainSteps:
- action: aws:downloadContent
  name: downloadContent
  inputs:
    sourceType: "{{ sourceType }}"
    sourceInfo: "{{ sourceInfo }}"
```

JSON

```
{
  "schemaVersion": "2.2",
  "description": "aws:downloadContent",
  "parameters": {
    "sourceType": {
    "description": "(Required) The download source.",
    "type": "String"
  },
  "sourceInfo": {
    "description": "(Required) The information required to retrieve the content from
 the required source.",
    "type": "StringMap"
    }
  },
  "mainSteps": [
    {
      "action": "aws:downloadContent",
      "name": "downloadContent",
      "inputs": {
        "sourceType":"{{ sourceType }}",
        "sourceInfo":"{{ sourceInfo }}"
      }
    }
  ]
```

```
}
```

Inputs

sourceType

La source de téléchargement prend Systems Manager actuellement en charge les types de source suivants pour le téléchargement de scripts et SSM de documents : `GitHub`, et `S3` .`SSMDocument`

Type: String (chaîne)

Obligatoire: Oui

sourceInfo

Informations obligatoires pour récupérer le contenu à partir de la source requise.

Type: StringMap

Obligatoire: Oui

Pour sourceType GitHub , spécifiez les informations suivantes :

- propriétaire : Propriétaire du référentiel.
- Référentiel : Le nom du référentiel.
- Chemin : Chemin d'accès au fichier ou au répertoire que vous souhaitez télécharger.
- getOptions: Options supplémentaires pour récupérer du contenu à partir d'une branche autre que principale ou d'une validation spécifique dans le référentiel. getOptionspeut être omis si vous utilisez la dernière validation dans la branche principale.

  Ce paramètre utilise le format suivant :

  - Branche :*`branch_name`*

    La valeur par défaut est `master`.

    Le `branch` paramètre n'est obligatoire que si votre document SSM est stocké dans une branche autre que `master` .
  - commitID:*`commitID`*

    La valeur par défaut est `head`.

    Pour utiliser la version de votre document SSM dans un commit autre que le dernier, spécifiez l'ID de validation complet. Exemple :

```
"getOptions": "commitID:bbc1ddb94...b76d3bEXAMPLE",
```

- tokenInfo: Le Systems Manager paramètre (un SecureString paramètre) où vous stockez vos informations de jeton d'GitHubaccès, au format `{{ssm-secure:`*`secure-string-token-name`*`}}` .

  > **Note**
  >
  > Ce `tokenInfo` champ est le seul champ du plug-in de document SSM qui prend en charge un SecureString paramètre. Les SecureString paramètres ne sont pas pris en charge pour les autres champs, ni pour les autres plug-ins de document SSM.

```
{
    "owner":"TestUser",
```

```
    "repository":"GitHubTest",
    "path":"scripts/python/test-script",
    "getOptions":"branch:master",
    "tokenInfo":"{{ssm-secure:secure-string-token}}"
}
```

Pour sourceType S3, spécifiez les éléments suivants :

- Chemin : URL du fichier ou du répertoire que vous souhaitez télécharger à partir d' Amazon S3 .

```
{
    "path": "https://s3.amazonaws.com/doc-example-bucket/powershell/
helloPowershell.ps1"
}
```

Pour sourceType SSMDocument , spécifiez l'une des valeurs suivantes :

- nom : Nom et version du document au format suivant : `name:version`. La version est facultative.

```
{
    "name": "Example-RunPowerShellScript:3"
}
```

- nom : ARN du document au format suivant : arn:aws:ssm:*region*:*account_id*:document/*document_name*

```
{
   "name":"arn:aws:ssm:us-east-2:3344556677:document/MySharedDoc"
}
```

destinationPath

Chemin local facultatif sur l'instance dans laquelle vous souhaitez télécharger le fichier. Si vous ne spécifiez pas de chemin, le contenu est téléchargé sur un chemin d'accès relatif à votre ID de commande.

Type: String (chaîne)

Obligatoire: Non

aws:psModule

Installez PowerShell des modules sur une instance EC2. Ce plug-in s'exécute uniquement sur les systèmes d'exploitation Windows Server. Pour plus d'informations, consultez la section Documents AWS Systems Manager (p. 1211).

Syntax

Schéma 2.2

YAML

```
---
schemaVersion: '2.2'
description: aws:psModule
parameters:
  source:
    description: "(Required) The URL or local path on the instance to the application
      .zip file."
```

```
    type: String
mainSteps:
- action: aws:psModule
  name: psModule
  inputs:
    source: "{{ source }}"
```

JSON

```
{
  "schemaVersion": "2.2",
  "description": "aws:psModule",
  "parameters": {
    "source": {
      "description": "(Required) The URL or local path on the instance to the
 application .zip file.",
      "type": "String"
    }
  },
  "mainSteps": [
    {
      "action": "aws:psModule",
      "name": "psModule",
      "inputs": {
        "source": "{{ source }}"
      }
    }
  ]
}
```

Schéma 1.2

YAML

```
---
runtimeConfig:
  aws:psModule:
    properties:
    - runCommand: "{{ commands }}"
      source: "{{ source }}"
      sourceHash: "{{ sourceHash }}"
      workingDirectory: "{{ workingDirectory }}"
      timeoutSeconds: "{{ executionTimeout }}"
```

JSON

```
{
   "runtimeConfig":{
      "aws:psModule":{
         "properties":[
            {
               "runCommand":"{{ commands }}",
               "source":"{{ source }}",
               "sourceHash":"{{ sourceHash }}",
               "workingDirectory":"{{ workingDirectory }}",
               "timeoutSeconds":"{{ executionTimeout }}"
            }
         ]
      }
   }
}
```

Properties

runCommand

Commande à exécuter une fois le module installé.PowerShell

Type: StringList

Obligatoire: Non

source

URL ou chemin d'accès sur l'instance du fichier `.zip` de l'application.

Type: String (chaîne)

Obligatoire: Oui

sourceHash

Hachage SHA256 du fichier `.zip`.

Type: String (chaîne)

Obligatoire: Non

timeoutSeconds

Nombre de secondes accordées à l'exécution d'une commande avant qu'elle soit considérée comme ayant échoué.

Type: String (chaîne)

Obligatoire: Non

workingDirectory

Chemin d'accès au répertoire de travail sur votre instance.

Type: String (chaîne)

Obligatoire: Non

aws:refreshAssociation

(Version de schéma 2.0 ou ultérieure) Actualisation (application forcée) d'une association à la demande. Cette action modifie l'état du système en fonction de ce qui est défini dans l'association sélectionnée ou dans toutes les associations liées aux cibles. Ce plug-in s'exécute sur les systèmes d'exploitation Linux et Microsoft Windows Server. Pour plus d'informations, consultez la section Documents AWS Systems Manager (p. 1211).

Syntax

Schéma 2.2

YAML

```
---
schemaVersion: '2.2'
description: aws:refreshAssociation
parameters:
  associationIds:
```

```
    description: "(Optional) List of association IDs. If empty, all associations bound
      to the specified target are applied."
    type: StringList
mainSteps:
- action: aws:refreshAssociation
  name: refreshAssociation
  inputs:
    associationIds:
    - "{{ associationIds }}"
```

JSON

```
{
  "schemaVersion": "2.2",
  "description": "aws:refreshAssociation",
  "parameters": {
    "associationIds": {
      "description": "(Optional) List of association IDs. If empty, all associations
 bound to the specified target are applied.",
      "type": "StringList"
    }
  },
  "mainSteps": [
    {
      "action": "aws:refreshAssociation",
      "name": "refreshAssociation",
      "inputs": {
        "associationIds": [
          "{{ associationIds }}"
        ]
      }
    }
  ]
}
```

Inputs

associationIds

Liste des associationsIDs. Si ce paramètre est vide, toutes les associations liées à la cible spécifiée sont appliquées.

Type: StringList

Obligatoire: Non

aws:runDockerAction

(Version de schéma 2.0 ou ultérieure) Exécution des actions Docker sur les conteneurs. Ce plug-in s'exécute sur les systèmes d'exploitation Linux et Microsoft Windows Server. Pour plus d'informations, consultez la section Documents AWS Systems Manager (p. 1211).

Syntax

Schéma 2.2

YAML

```
---
```

```
mainSteps:
- action: aws:runDockerAction
  name: RunDockerAction
  inputs:
    action: "{{ action }}"
    container: "{{ container }}"
    image: "{{ image }}"
    memory: "{{ memory }}"
    cpuShares: "{{ cpuShares }}"
    volume: "{{ volume }}"
    cmd: "{{ cmd }}"
    env: "{{ env }}"
    user: "{{ user }}"
    publish: "{{ publish }}"
```

JSON

```
{
   "mainSteps":[
      {
         "action":"aws:runDockerAction",
         "name":"RunDockerAction",
         "inputs":{
            "action":"{{ action }}",
            "container":"{{ container }}",
            "image":"{{ image }}",
            "memory":"{{ memory }}",
            "cpuShares":"{{ cpuShares }}",
            "volume":"{{ volume }}",
            "cmd":"{{ cmd }}",
            "env":"{{ env }}",
            "user":"{{ user }}",
            "publish":"{{ publish }}"
         }
      }
   ]
}
```

Inputs

action

Type d'action à effectuer.

Type: String (chaîne)

Obligatoire: Oui

conteneur

ID du conteneur Docker.

Type: String (chaîne)

Obligatoire: Non

image

Nom de l'image Docker.

Type: String (chaîne)

Obligatoire: Non

cmd

Commande du conteneur.

Type: String (chaîne)

Obligatoire: Non

memory

Limite de mémoire du conteneur.

Type: String (chaîne)

Obligatoire: Non

cpuShares

Parts d'UC du conteneur (poids relatif).

Type: String (chaîne)

Obligatoire: Non

volume

Montages de volume du conteneur.

Type: StringList

Obligatoire: Non

env

Variables d'environnement du conteneur.

Type: String (chaîne)

Obligatoire: Non

user

Nom d'utilisateur du conteneur.

Type: String (chaîne)

Obligatoire: Non

publish

Ports publiés du conteneur.

Type: String (chaîne)

Obligatoire: Non

aws:runDocument

(Version de schéma 2.0 ou ultérieure) Exécution des documents SSM stockés dans Systems Manager ou sur un partage local. Vous pouvez utiliser ce plug-in avec le plug-in aws:downloadContent (p. 1248) pour

télécharger un document SSM à partir d'un emplacement distant vers un partage local, puis l'exécuter. Ce plug-in est pris en charge sur les systèmes d'exploitation Linux et Windows Server.

Syntax

Schéma 2.2

YAML

```
---
schemaVersion: '2.2'
description: aws:runDocument
parameters:
  documentType:
    description: "(Required) The document type to run."
    type: String
    allowedValues:
    - LocalPath
    - SSMDocument
mainSteps:
- action: aws:runDocument
  name: runDocument
  inputs:
    documentType: "{{ documentType }}"
```

JSON

```
{
  "schemaVersion": "2.2",
  "description": "aws:runDocument",
  "parameters": {
    "documentType": {
      "description": "(Required) The document type to run.",
      "type": "String",
      "allowedValues": [
        "LocalPath",
        "SSMDocument"
      ]
    }
  },
  "mainSteps": [
    {
      "action": "aws:runDocument",
      "name": "runDocument",
      "inputs": {
        "documentType": "{{ documentType }}"
      }
    }
  ]
}
```

Inputs

documentType

Type du document à exécuter. Vous pouvez exécuter des documents locaux (`LocalPath`) ou des documents stockés dans Systems Manager (`SSMDocument`).

Type: String (chaîne)

Obligatoire: Oui

documentPath

Chemin d'accès du document. Si `documentType` est `LocalPath`, spécifiez le chemin d'accès au document sur le partage local. Si `documentType` est `SSMDocument`, spécifiez le nom du document.

Type: String (chaîne)

Obligatoire: Non

documentParameters

Paramètres pour le document.

Type: StringMap

Obligatoire: Non

aws:runPowerShellScript

Exécutez PowerShell des scripts ou spécifiez le chemin d'accès à un script à exécuter. Ce plug-in s'exécute sur les systèmes d'exploitation Linux et Microsoft Windows Server. Pour plus d'informations, consultez la section Documents AWS Systems Manager (p. 1211).

Syntax

Schéma 2.2

YAML

```
---
schemaVersion: '2.2'
description: aws:runPowerShellScript
parameters:
  commands:
    type: String
    description: "(Required) The commands to run or the path to an existing script
      on the instance."
    default: Write-Host "Hello World"
mainSteps:
- action: aws:runPowerShellScript
  name: runPowerShellScript
  inputs:
    timeoutSeconds: '60'
    runCommand:
    - "{{ commands }}"
```

JSON

```
{
  "schemaVersion": "2.2",
  "description": "aws:runPowerShellScript",
  "parameters": {
    "commands": {
      "type": "String",
      "description": "(Required) The commands to run or the path to an existing script
 on the instance.",
      "default": "Write-Host \"Hello World\""
    }
  },
  "mainSteps": [
```

```
    {
      "action": "aws:runPowerShellScript",
      "name": "runPowerShellScript",
      "inputs": {
        "timeoutSeconds": "60",
        "runCommand": [
          "{{ commands }}"
        ]
      }
    }
  ]
}
```

Schéma 1.2

YAML

```
---
runtimeConfig:
  aws:runPowerShellScript:
    properties:
    - id: 0.aws:runPowerShellScript
      runCommand: "{{ commands }}"
      workingDirectory: "{{ workingDirectory }}"
      timeoutSeconds: "{{ executionTimeout }}"
```

JSON

```
{
   "runtimeConfig":{
      "aws:runPowerShellScript":{
         "properties":[
            {
               "id":"0.aws:runPowerShellScript",
               "runCommand":"{{ commands }}",
               "workingDirectory":"{{ workingDirectory }}",
               "timeoutSeconds":"{{ executionTimeout }}"
            }
         ]
      }
   }
}
```

Properties

runCommand

Spécifiez les commandes à exécuter ou le chemin d'accès à un script existant sur l'instance.

Type: StringList

Obligatoire: Oui

timeoutSeconds

Nombre de secondes accordées à l'exécution d'une commande avant qu'elle soit considérée comme ayant échoué. Une fois le délai atteint, Systems Manager arrête l'exécution de la commande.

Type: String (chaîne)

Obligatoire: Non

workingDirectory

Chemin d'accès au répertoire de travail sur votre instance.

Type: String (chaîne)

Obligatoire: Non

aws : runShellScript

Exécution des scripts shell Linux ou spécification du chemin d'accès à un script à exécuter. Ce plug-in s'exécute uniquement sur les systèmes d'exploitation Linux. Pour plus d'informations, consultez la section Documents AWS Systems Manager (p. 1211).

Syntax

Schéma 2.2

YAML

```
---
schemaVersion: '2.2'
description: aws:runShellScript
parameters:
  commands:
    type: String
    description: "(Required) The commands to run or the path to an existing script
      on the instance."
    default: echo Hello World
mainSteps:
- action: aws:runShellScript
  name: runShellScript
  inputs:
    timeoutSeconds: '60'
    runCommand:
    - "{{ commands }}"
```

JSON

```
{
  "schemaVersion": "2.2",
  "description": "aws:runShellScript",
  "parameters": {
    "commands": {
      "type": "String",
      "description": "(Required) The commands to run or the path to an existing script
 on the instance.",
      "default": "echo Hello World"
    }
  },
  "mainSteps": [
    {
      "action": "aws:runShellScript",
      "name": "runShellScript",
      "inputs": {
        "timeoutSeconds": "60",
        "runCommand": [
          "{{ commands }}"
```

```
        ]
      }
    }
  ]
}
```

Schéma 1.2

YAML

```
---
runtimeConfig:
  aws:runShellScript:
    properties:
    - runCommand: "{{ commands }}"
      workingDirectory: "{{ workingDirectory }}"
      timeoutSeconds: "{{ executionTimeout }}"
```

JSON

```
{
   "runtimeConfig":{
      "aws:runShellScript":{
         "properties":[
            {
               "runCommand":"{{ commands }}",
               "workingDirectory":"{{ workingDirectory }}",
               "timeoutSeconds":"{{ executionTimeout }}"
            }
         ]
      }
   }
}
```

Properties

runCommand

Spécifiez les commandes à exécuter ou le chemin d'accès à un script existant sur l'instance.

Type: StringList

Obligatoire: Oui

timeoutSeconds

Nombre de secondes accordées à l'exécution d'une commande avant qu'elle soit considérée comme ayant échoué. Une fois le délai atteint, Systems Manager arrête l'exécution de la commande.

Type: String (chaîne)

Obligatoire: Non

workingDirectory

Chemin d'accès au répertoire de travail sur votre instance.

Type: String (chaîne)

Obligatoire: Non

aws:softwareInventory

(Version de schéma 2.0 ou version ultérieure) Recueil des métadonnées sur des applications, des fichiers et des configurations sur vos instances gérées. Ce plug-in s'exécute sur les systèmes d'exploitation Linux et Microsoft Windows Server. Lorsque vous configurez la collecte d'inventaire, vous commencez par créer une association Systems Manager State Manager. Systems Manager collecte les données d'inventaire lorsque cette association est exécutée. Si vous ne créez pas d'abord l'association et essayez d'appeler le plug-in aws:softwareInventory, le système renvoie l'erreur suivante :

```
The aws:softwareInventory plugin can only be invoked via ssm-associate.
```

Notez également qu'une instance peut avoir uniquement une association d'inventaire configurée en même temps. Si vous configurez une instance avec plusieurs associations, l'association d'inventaire n'est pas exécutée et aucune donnée d'inventaire n'est collectée. Pour de plus amples informations sur la collecte de l'inventaire, veuillez consulter Inventaire AWS Systems Manager (p. 791).

Syntax

Schéma 2.2

YAML

```
---
mainSteps:
- action: aws:softwareInventory
  name: collectSoftwareInventoryItems
  inputs:
    applications: "{{ applications }}"
    awsComponents: "{{ awsComponents }}"
    networkConfig: "{{ networkConfig }}"
    files: "{{ files }}"
    services: "{{ services }}"
    windowsRoles: "{{ windowsRoles }}"
    windowsRegistry: "{{ windowsRegistry}}"
    windowsUpdates: "{{ windowsUpdates }}"
    instanceDetailedInformation: "{{ instanceDetailedInformation }}"
    customInventory: "{{ customInventory }}"
```

JSON

```
{
   "mainSteps":[
      {
         "action":"aws:softwareInventory",
         "name":"collectSoftwareInventoryItems",
         "inputs":{
            "applications":"{{ applications }}",
            "awsComponents":"{{ awsComponents }}",
            "networkConfig":"{{ networkConfig }}",
            "files":"{{ files }}",
            "services":"{{ services }}",
            "windowsRoles":"{{ windowsRoles }}",
            "windowsRegistry":"{{ windowsRegistry}}",
            "windowsUpdates":"{{ windowsUpdates }}",
            "instanceDetailedInformation":"{{ instanceDetailedInformation }}",
            "customInventory":"{{ customInventory }}"
```

```
            }
        }
    ]
}
```

Inputs

applications

(Facultatif) Collecter les métadonnées pour les applications installées.

Type: String (chaîne)

Obligatoire: Non

awsComponents

Description : (Facultatif) collecte de métadonnées pour des composants AWS tels qu'amazon-ssm-agent.

Type: String (chaîne)

Obligatoire: Non

fichiers

(Facultatif, nécessite la version de l'agent SSM 2.2.64.0 ou d'une version ultérieure) Collectez les métadonnées pour les fichiers, notamment leurs noms, leur heure de création, l'heure de leur dernière modification et de leur dernier accès, leur taille, etc. Pour de plus amples informations sur la collecte de l'inventaire de fichiers, veuillez consulter Utilisation de l'inventaire de fichiers et du registre Windows (p. 798).

Type: String (chaîne)

Obligatoire: Non

networkConfig

(Facultatif) Collectez les métadonnées pour les configurations de réseau.

Type: String (chaîne)

Obligatoire: Non

windowsUpdates

(Facultatif) Collectez les métadonnées pour toutes les mises à jour Windows.

Type: String (chaîne)

Obligatoire: Non

instanceDetailedInformation

(Facultatif) Collectez plus d'informations d'instance que ce qui est fourni par le plug-in d'inventaire par défaut (aws:instanceInformation), y compris le modèle d'UC, la vitesse et le nombre de cœurs, pour n'en citer que quelques-uns.

Type: String (chaîne)

Obligatoire: Non

services

(Facultatif, système d'exploitation, Windows nécessite uniquement la version de l'agent SSM 2.2.64.0 ou une version ultérieure) Collectez les métadonnées pour les configurations de service.

Type: String (chaîne)

Obligatoire: Non

windowsRegistry

(Facultatif, système d'exploitation, Windows nécessite uniquement la version de l'agent SSM 2.2.64.0 ou une version ultérieure) Collectez les valeurs et clés de registre Windows. Vous pouvez choisir un chemin de clé et collecter toutes les clés et valeurs de manière récursive. Vous pouvez également collecter une clé de registre spécifique et sa valeur pour un chemin donné. Inventory collecte le chemin de clé, le nom, le type et la valeur. Pour de plus amples informations sur la collecte de l'inventaire de fichiers et du registre Windows, veuillez consulter Utilisation de l'inventaire de fichiers et du registre Windows (p. 798).

Type: String (chaîne)

Obligatoire: Non

windowsRoles

(Facultatif, système d'exploitation, Windows nécessite uniquement la version de l'agent SSM 2.2.64.0 ou une version ultérieure) Collectez les métadonnées pour les configurations de rôle Microsoft Windows.

Type: String (chaîne)

Obligatoire: Non

customInventory

(Facultatif) Collectez les données d'inventaire personnalisées. Pour de plus amples informations sur l'inventaire personnalisé, veuillez consulter Utilisation de l'inventaire personnalisé (p. 826).

Type: String (chaîne)

Obligatoire: Non

aws:updateAgent

Mettez à jour le EC2Config service vers la dernière version ou spécifiez une version plus ancienne. Ce plug-in s'exécute uniquement sur les systèmes d'exploitation Microsoft Windows Server. Pour plus d'informations sur le EC2Config service , consultez Configuration d'une instance Windows à l'aide du EC2Config service . Pour plus d'informations sur les documents, consultez Documents AWS Systems Manager (p. 1211).

Syntax

Schéma 2.2

YAML

```
---
schemaVersion: '2.2'
description: aws:updateAgent
mainSteps:
- action: aws:updateAgent
  name: updateAgent
```

```
  inputs:
    agentName: Ec2Config
    source: https://s3.{Region}.amazonaws.com/aws-ssm-{Region}/manifest.json
```

JSON

```
{
  "schemaVersion": "2.2",
  "description": "aws:updateAgent",
  "mainSteps": [
    {
      "action": "aws:updateAgent",
      "name": "updateAgent",
      "inputs": {
        "agentName": "Ec2Config",
        "source": "https://s3.{Region}.amazonaws.com/aws-ssm-{Region}/manifest.json"
      }
    }
  ]
}
```

Schéma 1.2

YAML

```
---
runtimeConfig:
  aws:updateAgent:
    properties:
      agentName: Ec2Config
      source: https://s3.{Region}.amazonaws.com/aws-ssm-{Region}/manifest.json
      allowDowngrade: "{{ allowDowngrade }}"
      targetVersion: "{{ version }}"
```

JSON

```
{
   "runtimeConfig":{
      "aws:updateAgent":{
         "properties":{
            "agentName":"Ec2Config",
            "source":"https://s3.{Region}.amazonaws.com/aws-ssm-{Region}/
manifest.json",
            "allowDowngrade":"{{ allowDowngrade }}",
            "targetVersion":"{{ version }}"
         }
      }
   }
}
```

Properties

agentName

EC2Config. Il s'agit du nom de l'agent qui exécute le EC2Config service.

Type: String (chaîne)

Obligatoire: Oui

allowDowngrade

Permet au EC2Config service d'être dégradé vers une version antérieure. Si false est défini, le service peut uniquement être mis à jour vers des versions plus récentes (valeur par défaut). Si true est défini, vous devez spécifier la version antérieure.

Type: Booléen

Obligatoire: Non

source

Emplacement où Systems Manager copie la version d' EC2Config à installer. Vous ne pouvez pas modifier cet emplacement.

Type: String (chaîne)

Obligatoire: Oui

targetVersion

Version spécifique du EC2Config service à installer. Si vous ne spécifiez pas de version, le service est mis à jour avec la dernière version.

Type: String (chaîne)

Obligatoire: Non

aws:updateSsmAgent

Mettez à jour l' Agent SSM vers la dernière version ou spécifiez une version plus ancienne. Ce plug-in s'exécute sur les systèmes d'exploitation Linux et Windows Server. Pour plus d'informations, consultez la section Utilisation des Agent SSM (p. 72). Pour plus d'informations sur les documents, consultez Documents AWS Systems Manager (p. 1211).

Syntax

Schéma 2.2

YAML

```
---
schemaVersion: '2.2'
description: aws:updateSsmAgent
parameters:
  allowDowngrade:
    default: 'false'
    description: "(Optional) Allow the Amazon SSM Agent service to be downgraded to
      an earlier version. If set to false, the service can be upgraded to newer
 versions
      only (default). If set to true, specify the earlier version."
    type: String
    allowedValues:
    - 'true'
    - 'false'
mainSteps:
- action: aws:updateSsmAgent
  name: updateSSMAgent
  inputs:
    agentName: amazon-ssm-agent
    source: https://s3.{Region}.amazonaws.com/amazon-ssm-{Region}/ssm-agent-
manifest.json
```

```
    allowDowngrade: "{{ allowDowngrade }}"
```

JSON

```
{
  "schemaVersion": "2.2",
  "description": "aws:updateSsmAgent",
  "parameters": {
    "allowDowngrade": {
      "default": "false",
      "description": "(Required) Allow the Amazon SSM Agent service to be downgraded
 to an earlier version. If set to false, the service can be upgraded to newer versions
 only (default). If set to true, specify the earlier version.",
      "type": "String",
      "allowedValues": [
        "true",
        "false"
      ]
    }
  },
  "mainSteps": [
    {
      "action": "aws:updateSsmAgent",
      "name": "updateSSMAgent",
      "inputs": {
        "agentName": "amazon-ssm-agent",
        "source": "https://s3.{Region}.amazonaws.com/amazon-ssm-{Region}/ssm-agent-
manifest.json",
        "allowDowngrade": "{{ allowDowngrade }}"
      }
    }
  ]
}
```

Schéma 1.2

YAML

```
---
runtimeConfig:
  aws:updateSsmAgent:
    properties:
    - agentName: amazon-ssm-agent
      source: https://s3.{Region}.amazonaws.com/aws-ssm-{Region}/manifest.json
      allowDowngrade: "{{ allowDowngrade }}"
```

JSON

```
{
   "runtimeConfig":{
      "aws:updateSsmAgent":{
         "properties":[
            {
               "agentName":"amazon-ssm-agent",
               "source":"https://s3.{Region}.amazonaws.com/aws-ssm-{Region}/
manifest.json",
               "allowDowngrade":"{{ allowDowngrade }}"
            }
         ]
      }
   }
```

```
}
```

Properties

agentName

amazon-ssm-agent. Nom de l'agent Systems Manager qui traite les demandes et exécute les commandes sur l'instance.

Type: String (chaîne)

Obligatoire: Oui

allowDowngrade

Permet Agent SSM à d'être dégradé vers une version antérieure. Si false est défini, l'agent peut uniquement être mis à jour vers des versions plus récentes (valeur par défaut). Si true est défini, vous devez spécifier la version antérieure.

Type: Booléen

Obligatoire: Oui

source

Emplacement où Systems Manager copie la version de l'Agent SSM à installer. Vous ne pouvez pas modifier cet emplacement.

Type: String (chaîne)

Obligatoire: Oui

targetVersion

Version spécifique de Agent SSM à installer. Si vous ne spécifiez pas de version, l'agent est mis à jour avec la dernière version.

Type: String (chaîne)

Obligatoire: Non

Création de documents Systems Manager

Si les documents publics Systems Manager n'exécutent pas toutes les actions que vous souhaitez effectuer sur vos ressources AWS, vous pouvez créer vos propres documents SSM. Lorsque vous créez un nouveau `Command` ou document `Policy`, nous vous recommandons d'utiliser la version 2.2 ou ultérieure du schéma afin de tirer parti des fonctionnalités les plus récentes, telles que la mise à jour de documents, le contrôle de version automatique, le séquençage, etc.

Rédaction du contenu du document SSM

Pour créer votre propre contenu de document SSM, il est important de comprendre les différents schémas, fonctionnalités, plug-ins et syntaxe disponibles pour les documents SSM. Nous vous recommandons de vous familiariser avec les ressources suivantes.

- Écriture de vos propres AWS Systems Manager documents
- Syntaxe du document SSM (p. 1226)
- Schémas et fonctionnalités de document SSM (p. 1213)
- Référence du plug-in de document Systems Manager (p. 1232)

- Référence sur les actions Systems Manager Automation (p. 386)
- Variables système Automation (p. 437)
- Exemples de scénarios et solutions de documents Automation personnalisées (p. 481)
- Utilisation des documents Systems Manager Automation à l'aide de l'AWS Toolkit pour Visual Studio Code
- Procédure : Utilisation de Document Builder pour créer un document Automation personnalisé (p. 670)
- Création de documents Automation qui exécutent des scripts (p. 451)

Les documents SSM prédéfinis AWS peuvent effectuer certaines des actions dont vous avez besoin. Vous pouvez appeler ces documents en utilisant les plug-ins `aws:runDocument`, `aws:runCommand` ou `aws:executeAutomation` dans votre document personnalisé SSM, selon le type de document. Vous pouvez également copier des parties de ces documents dans un document personnalisé SSM et modifier le contenu pour répondre à vos besoins.

> **Tip**
>
> Lors de la création du contenu du document SSM, vous pouvez modifier le contenu et mettre à jour le document SSM plusieurs fois pendant le test. Les commandes suivantes mettent à jour le document SSM avec votre dernier contenu et mettent à jour la version par défaut du document vers la dernière version du document.
>
> > **Note**
> >
> > Les commandes Linux et Windows utilisent l'outil de ligne de commande `jq` pour filtrer les données de réponse JSON.
>
> Linux

```
latestDocVersion=$(aws ssm update-document \
    --content file://path/to/file/documentContent.json \
    --name "ExampleDocument" \
    --document-format JSON \
    --document-version '$LATEST' \
    | jq -r '.DocumentDescription.LatestVersion')

aws ssm update-document-default-version \
    --name "ExampleDocument" \
    --document-version $latestDocVersion
```

> Windows

```
latestDocVersion=$(aws ssm update-document ^
    --content file://C:\path\to\file\documentContent.json ^
    --name "ExampleDocument" ^
    --document-format JSON ^
    --document-version "$LATEST" ^
    | jq -r '.DocumentDescription.LatestVersion')

aws ssm update-document-default-version ^
    --name "ExampleDocument" ^
    --document-version $latestDocVersion
```

> PowerShell

```
$content = Get-Content -Path "C:\path\to\file\documentContent.json" | Out-String
$latestDocVersion = Update-SSMDocument `
    -Content $content `
    -Name "ExampleDocument" `
    -DocumentFormat "JSON" `
```

```
    -DocumentVersion '$LATEST' `
    | Select-Object -ExpandProperty LatestVersion

Update-SSMDocumentDefaultVersion `
    -Name "ExampleDocument" `
    -DocumentVersion $latestDocVersion
```

Utilisation de documents SSM dans les associations State Manager

Si vous créez un document SSM pour State Manager, vous devez l'associer à vos instances gérées après l'avoir ajouté dans le système. Pour plus d'informations, consultez la section Créer une association (p. 993).

Gardez à l'esprit les détails suivants lorsque vous utilisez des documents SSM dans des associations State Manager.

- Vous pouvez attribuer plusieurs documents à une cible en créant différentes associations State Manager qui utilisent différents documents.
- Si vous créez un document avec des plug-ins contradictoires (par exemple, joindre au domaine et supprimer du domaine), c'est le dernier plug-in exécuté qui correspondra à l'état final. State Manager ne valide pas la séquence logique ou la cohérence des commandes ou des plug-ins dans votre document.
- Lors du traitement des documents, les associations d'instances sont appliquées en premier, puis vient le tour des associations de groupes balisés. Si une instance fait partie de plusieurs groupes balisés, les documents appartenant au groupe balisé ne seront pas exécutés dans un ordre spécifique. Si une instance est directement ciblée par le biais de plusieurs documents en fonction de son ID d'instance, l'exécution se déroule dans un ordre aléatoire.
- Si vous modifiez la version par défaut d'un document SSM pour State Manager, toute association ayant recours à ce document commencera à utiliser la nouvelle version par défaut la prochaine fois que Systems Manager appliquera l'association à l'instance.
- Si vous créez une association à l'aide d'un SSM document qui a été partagé avec vous, puis le propriétaire s'arrête de partager le document avec vous, vos associations n'a plus accès à ce document. Toutefois, si le propriétaire partage le même SSM document avec vous ultérieurement, vos associations remapper automatiquement à celui-ci.

Après avoir écrit le contenu du document SSM, vous pouvez utiliser votre contenu pour créer un document SSM à l'aide de l'une des méthodes suivantes.

Créer des documents SSM

- Créer un document SSM (console) (p. 1269)
- Créer un document SSM (ligne de commande) (p. 1270)
- Créer un document SSM (API) (p. 1271)
- Création de documents composites (p. 1273)

Créer un document SSM (console)

Après avoir créé le contenu pour votre document SSM personnalisé, comme décrit dans Rédaction du contenu du document SSM (p. 1267), vous pouvez utiliser la console Systems Manager pour créer un document SSM à l'aide de votre contenu.

Pour créer un document SSM (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.

2. Dans le panneau de navigation, cliquez sur Documents.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Documents dans le panneau de navigation.
3. Choisissez Create command or session (Créer une commande ou une session).
4. Tapez un nom descriptif pour le document
5. (Facultatif) Pour Type de cible, spécifiez le type de ressources sur lequel le document peut s'exécuter.
6. Dans la liste Types de document, sélectionnez le type de document que vous souhaitez créer.
7. Supprimez les crochets du champ Contenu, puis copiez et collez le document que vous avez créé précédemment.
8. (Facultatif) Dans la section Balises du document appliquez une ou plusieurs paires nom/valeur de clé de balise au document.

   Les balises sont des métadonnées facultatives que vous affectez à une ressource. Les balises vous permettent de classer une ressource de différentes façons, par exemple, par objectif, par propriétaire ou par environnement. Par exemple, vous pouvez baliser un document pour identifier le type de tâches qu'il exécute, le type de systèmes d'exploitation qu'il cible et l'environnement dans lequel il s'exécute. Dans ce cas, vous pouvez spécifier les paires nom/valeur de clé suivantes :

   - `Key=TaskType,Value=MyConfigurationUpdate`
   - `Key=OS,Value=AMAZON_LINUX_2`
   - `Key=Environment,Value=Production`

   Pour de plus amples informations sur le balisage des ressources Systems Manager, veuillez consulter Balisage des ressources Systems Manager (p. 1359).
9. Choisissez Create document (Créer un document) pour enregistrer le document.

Créer un document SSM (ligne de commande)

Après avoir créé le contenu pour votre document SSM personnalisé, comme décrit dans Rédaction du contenu du document SSM (p. 1267), vous pouvez utiliser AWS CLI ou Outils AWS pour PowerShell pour créer un document SSM à l'aide de votre contenu. En voici un exemple dans la commande suivante :

Avant de commencer

Installez et configurez l'AWS CLI ou Outils AWS pour PowerShell si vous ne l'avez pas déjà fait. Pour plus d'informations, consultez la section Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).

Linux

```
aws ssm create-document \
    --content file://path/to/file/documentContent.json \
    --name "document-name" \
    --document-type "Command" \
    --tags "Key=tag-key,Value=tag-value"
```

Windows

```
aws ssm create-document ^
    --content file://C:\path\to\file\documentContent.json ^
```

```
    --name "document-name" ^
    --document-type "Command" ^
    --tags "Key=tag-key,Value=tag-value"
```

PowerShell

```
$json = Get-Content -Path "C:\path\to\file\documentContent.json" | Out-String
New-SSMDocument `
    -Content $json `
    -Name "document-name" `
    -DocumentType "Command" `
    -Tags "Key=tag-key,Value=tag-value"
```

document-name est le nom du SSM document que vous souhaitez baliser.

tag-key est le nom d'une clé personnalisée que vous fournissez. Par exemple, région ou quartier.

tag-value est le contenu personnalisé pour la valeur que vous souhaitez fournir pour cette clé. Par exemple, Ouestou Q321.

Si elle aboutit, la commande renvoie une réponse semblable à la suivante :

```
{
   "DocumentDescription":{
      "CreatedDate":1.585061751738E9,
      "DefaultVersion":"1",
      "Description":"MyCustomDocument",
      "DocumentFormat":"JSON",
      "DocumentType":"Command",
      "DocumentVersion":"1",
      "Hash":"0d3d879b3ca072e03c12638d0255ebd004d2c65bd318f8354fcde820dEXAMPLE",
      "HashType":"Sha256",
      "LatestVersion":"1",
      "Name":"Example",
      "Owner":"111122223333",
      "Parameters":[
         --truncated--
      ],
      "PlatformTypes":[
         "Windows",
         "Linux"
      ],
      "SchemaVersion":"0.3",
      "Status":"Creating",
      "Tags": [
            {
                "Key": "Purpose",
                "Value": "Test"
            }
        ]
   }
}
```

Créer un document SSM (API)

Une fois que vous avez créé le contenu de votre SSM document personnalisé, comme décrit dans Rédaction du contenu du document SSM (p. 1267) , vous pouvez utiliser votre kit SDK préféré pour appeler lAWS Systems ManagerCreateDocument'action d'API afin de créer un SSM document à l'aide de votre contenu. La chaîne JSON ou YAML du paramètre de demande `Content` est généralement lue à

partir d'un fichier. Les exemples de fonctions suivants créent un SSM document à l'aide du SDKs kit pour Python, Go et Java.

Python

```
import boto3

ssm = boto3.client('ssm')
filepath = '/path/to/file/documentContent.yaml'

def createDocumentApiExample():
    with open(filepath) as openFile:
        documentContent = openFile.read()
        createDocRequest = ssm.create_document(
            Content = documentContent,
            Name = 'createDocumentApiExample',
            DocumentType = 'Automation',
            DocumentFormat = 'YAML'
        )
        print(createDocRequest)

createDocumentApiExample()
```

Go

```
package main

import (
    "github.com/aws/aws-sdk-go/aws"
    "github.com/aws/aws-sdk-go/aws/session"
    "github.com/aws/aws-sdk-go/service/ssm"

    "fmt"
    "io/ioutil"
    "log"
)


func main() {
    openFile, err := ioutil.ReadFile("/path/to/file/documentContent.yaml")
    if err != nil {
  log.Fatal(err)
 }
    documentContent := string(openFile)
    sesh := session.Must(session.NewSessionWithOptions(session.Options{
        SharedConfigState: session.SharedConfigEnable}))

    ssmClient := ssm.New(sesh)
    createDocRequest, err := ssmClient.CreateDocument(&ssm.CreateDocumentInput{
        Content: &documentContent,
        Name:  aws.String("createDocumentApiExample"),
        DocumentType: aws.String("Automation"),
        DocumentFormat: aws.String("YAML"),
    })
    result := *createDocRequest
    fmt.Println(result)
}
```

Java

```
import java.io.IOException;
import java.nio.charset.Charset;
import java.nio.charset.StandardCharsets;
import java.nio.file.Files;
import java.nio.file.Paths;

import com.amazonaws.AmazonClientException;
import com.amazonaws.AmazonServiceException;
import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.regions.Regions;
import com.amazonaws.services.simplesystemsmanagement.AWSSimpleSystemsManagement;
import
 com.amazonaws.services.simplesystemsmanagement.AWSSimpleSystemsManagementClientBuilder;
import com.amazonaws.services.simplesystemsmanagement.model.*;

public class createDocumentApiExample {
  public static void main(String[] args) {
    try {
      createDocumentMethod(getDocumentContent());
    }
    catch (IOException e) {
      e.printStackTrace();
    }
  }
    public static String getDocumentContent() throws IOException {
      String filepath = new String("/path/to/file/documentContent.yaml");
        byte[] encoded = Files.readAllBytes(Paths.get(filepath));
        String documentContent = new String(encoded, StandardCharsets.UTF_8);
        return documentContent;
    }

    public static void createDocumentMethod (final String documentContent) {
      AWSSimpleSystemsManagement ssm =
 AWSSimpleSystemsManagementClientBuilder.defaultClient();
      final CreateDocumentRequest createDocRequest = new CreateDocumentRequest()
        .withContent(documentContent)
        .withName("createDocumentApiExample")
        .withDocumentType("Automation")
        .withDocumentFormat("YAML");
      final CreateDocumentResult result = ssm.createDocument(createDocRequest);
    }
}
```

Pour plus d'informations sur la création de contenu de document personnalisé, veuillez consulter Syntaxe du document SSM (p. 1226).

Création de documents composites

Un document SSM composite est un document personnalisé qui effectue une série d'actions en exécutant un ou plusieurs documents SSM secondaires. Les documents composites favorisent l'infrastructure en tant que code en vous permettant de créer un jeu standard de documents SSM pour des tâches courantes telles que l'amorçage de logiciel ou la jonction d'instances à un domaine. Vous pouvez ensuite partager ces documents entre plusieurs comptes AWS de la même région AWS afin de réduire la maintenance de documents SSM et d'assurer une cohérence.

Par exemple, vous pouvez créer un document composite qui effectue les actions suivantes :

1. Mettre à jour l'Agent SSM avec la version la plus récente.
2. Installer tous les correctifs sur liste blanche.
3. Installer un logiciel antivirus.

4. Télécharge des scripts depuis GitHub et les exécute.

Dans cet exemple, votre document SSM personnalisé inclut les plug-ins suivants pour effectuer ces actions :

1. Le plug-in `aws:runDocument` pour exécuter le document `AWS-UpdateSSMAgent` qui met à jour l'Agent SSM avec la version la plus récente.
2. Plug-in `aws:runDocument` pour exécuter le document AWS-ApplyPatchBaseline, qui installe tous les correctifs sur liste blanche.
3. Le plug-in `aws:runDocument` pour exécuter le document AWS-InstallApplication, qui installe le logiciel antivirus.
4. Plug`aws:downloadContent`-in à partir duquel télécharger des scripts GitHub et les exécuter.

Les documents composites et secondaires peuvent être stockés dans Systems Manager , GitHub (référentiels publics et privés) ou Amazon S3 dans . Les documents composites et secondaires peuvent être créés au format JSON ou YAML.

> **Note**
>
> Les documents composites peuvent exécuter une profondeur maximale de trois documents. Cela signifie qu'un document composite peut appeler un document enfant, et que celui-ci peut appeler un dernier document.

Créer un document composite

Pour créer un document composite, ajoutez le plug-in aws:runDocument (p. 1255) dans un document SSM personnalisé et spécifiez les entrées obligatoires. L'exemple suivant présente un document composite qui effectue les actions suivantes :

1. Exécute le aws:downloadContent (p. 1248) plug-in pour télécharger un SSM document à partir d'un référentiel GitHub public vers un répertoire local appelé amorçage. Le SSM document est appelé StateManagerBootstrap .yml (document YAML).
2. Exécute le `aws:runDocument` plug-in pour exécuter le document StateManagerBootstrap .yml. Aucun paramètre n'est spécifié.
3. Exécute le plug-in `aws:runDocument` pour exécuter le document SSM prédéfini AWS-ConfigureDocker. Les paramètres spécifiés installent Docker sur l'instance.

```
{
  "schemaVersion": "2.2",
  "description": "My composite document for bootstrapping software and installing Docker.",
  "parameters": {
  },
  "mainSteps": [
    {
      "action": "aws:downloadContent",
      "name": "downloadContent",
      "inputs": {
        "sourceType": "GitHub",
        "sourceInfo": "{\"owner\":\"TestUser1\",\"repository\":\"TestPublic\", \"path\":
\"documents/bootstrap/StateManagerBootstrap.yml\"}",
        "destinationPath": "bootstrap"
      }
    },
    {
      "action": "aws:runDocument",
      "name": "runDocument",
      "inputs": {
```

```
        "documentType": "LocalPath",
        "documentPath": "bootstrap",
        "documentParameters": "{}"
      }
    },
    {
      "action": "aws:runDocument",
      "name": "configureDocker",
      "inputs": {
        "documentType": "SSMDocument",
        "documentPath": "AWS-ConfigureDocker",
        "documentParameters": "{\"action\":\"Install\"}"
      }
    }
  ]
}
```

Voir aussi

- Pour de plus amples informations sur le redémarrage des serveurs et des instances en appelant des scripts à l'aide de Fonctionnalité Exécuter la commande, veuillez consulter Redémarrage d'une instance gérée à partir de scripts (p. 963).
- Pour plus d'informations sur la création d'un document SSM, consultez Création de documents Systems Manager (p. 1267).
- Pour plus d'informations sur les plug-ins que vous pouvez ajouter à un document SSM personnalisé, consultez Référence du plug-in de document Systems Manager (p. 1232).
- Si vous souhaitez simplement exécuter un document à partir d'un emplacement distant (sans créer de document composite), consultez Exécution de documents Systems Manager de commande à partir d'emplacements distants (p. 1286).

Partage de documents SSM

Vous pouvez partager les documents SSM en privé ou en public avec des comptes de la même région AWS. Pour partager un document de façon privée, modifiez les autorisations du document et autorisez des personnes spécifiques à y accéder en fonction de leur ID Amazon Web Services (AWS). Pour partager un document SSM publiquement, modifiez les autorisations du document et spécifiez `All`.

> **Warning**
>
> N'utilisez que des documents SSM partagés provenant de sources fiables. Lorsque vous utilisez un document partagé, étudiez soigneusement son contenu avant de l'utiliser afin de comprendre comment il va modifier la configuration de votre instance. Pour plus d'informations sur les bonnes pratiques en matière de documents partagés, consultez Bonnes pratiques pour les documents SSM partagés (p. 1276).

Limitations

Lorsque vous commencez à utiliser les documents SSM, vous devez connaître les limitations suivantes.

- Seul le propriétaire peut partager un document.
- Vous devez arrêter le partage d'un document pour pouvoir le supprimer. Pour plus d'informations, consultez la section Modification des autorisations d'un document SSM partagé (p. 1280).
- Vous pouvez partager un document avec 1 000 comptes AWS au maximum. Vous pouvez demander que cette limite soit augmentée dans le Centre AWS Support. Pour Limit type (Type de limite), choisissez EC2 Systems Manager et décrivez le motif de votre demande.
- Vous pouvez partager publiquement cinq documents SSM au maximum. Vous pouvez demander que cette limite soit augmentée dans le Centre AWS Support. Pour Limit type (Type de limite), choisissez EC2 Systems Manager et décrivez le motif de votre demande.

- Les documents peuvent être partagés avec d'autres comptes de la même région AWS uniquement. Le partage inter-régions n'est pas pris en charge.

Pour de plus amples informations sur les quotas de service d'Systems Manager, veuillez consulter Quotas de service AWS Systems Manager.

Table des matières

- Bonnes pratiques pour les documents SSM partagés (p. 1276)
- Partager un document SSM (p. 1277)
- Modification des autorisations d'un document SSM partagé (p. 1280)
- Utilisation de documents SSM partagés (p. 1281)

Bonnes pratiques pour les documents SSM partagés

Passez en revue les consignes suivantes avant de partager un document ou d'utiliser un document partagé.

Supprimer les informations sensibles

Étudiez soigneusement votre document SSM et supprimez les informations sensibles. Par exemple, vérifiez que le document n'inclut pas vos informations d'identification AWS. Si vous partagez un document avec des personnes spécifiques, ces personnes peuvent voir les informations contenues dans le document. Si vous partagez un document publiquement, tout le monde peut voir les informations contenues dans le document.

Limiter les actions Fonctionnalité Exécuter la commande à l'aide d'une stratégie d'approbation d'utilisateur IAM

Créez une stratégie d'utilisateur AWS Identity and Access Management (IAM) restrictive pour les utilisateurs qui auront accès au document. La stratégie IAM détermine les documents SSM qu'un utilisateur peut visualiser dans la console Amazon EC2 ou en appelant `ListDocuments` à l'aide de l'AWS CLI ou d'Outils AWS pour Windows PowerShell. La stratégie limite également les actions que l'utilisateur peut effectuer avec un document SSM. Vous pouvez créer une stratégie restrictive pour qu'un utilisateur ne puisse utiliser que des documents spécifiques. Pour de plus amples informations, veuillez consulter Créer des utilisateurs et des groupes IAM non administrateurs pour Systems Manager (p. 30) et Exemples de stratégies gérées par le client (p. 1307).

Faites preuve de prudence lors de l'utilisation de documents SSM partagés

Étudiez le contenu de chaque document partagé avec vous, notamment des documents publics, afin de comprendre les commandes qui seront exécutées sur vos instances. Un document pourrait avoir, volontairement ou non, des répercussions négatives après son exécution. Si le document fait référence à un réseau externe, étudiez la source externe avant d'utiliser ce document.

Envoyer les commandes en utilisant le hachage de document

Lorsque vous partagez un document, le système crée un hachage Sha-256 et l'affecte au document. Le système enregistre également un instantané du contenu du document. Lorsque vous envoyez une commande à l'aide d'un document partagé, vous pouvez spécifier le hachage dans votre commande afin de garantir le respect des conditions suivantes :

- Vous exécutez la commande à partir du document Systems Manager correct.
- Le contenu du document n'a pas changé depuis qu'il a été partagé avec vous.

Si le hachage ne correspond pas au document spécifié ou si le contenu du document partagé a changé, la commande renvoie une exception `InvalidDocument`. Remarque: Le hachage ne peut pas vérifier le contenu du document à partir d'emplacements externes.

Partager un document SSM

Vous pouvez partager les documents SSM à l'aide de la console AWS Systems Manager. Vous pouvez également partager des documents SSM par programme en appelant l'action d'API `ModifyDocumentPermission` à l'aide de l'AWS CLI, d'Outils AWS pour Windows PowerShell ou du kit SDK AWS. Avant de partager un document, obtenez le compte AWS IDs des personnes avec lesquelles vous souhaitez le partager. Vous spécifierez ces comptes IDs lorsque vous partagerez le document.

Rubriques

- Partager un document (console) (p. 1277)
- Partager un document (ligne de commande) (p. 1277)

Partager un document (console)

Partage d'un document

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Documents dans le panneau de navigation.
3. Dans la liste des documents, sélectionnez le document que vous souhaitez partager, puis choisissez Afficher les détails. Sur l'onglet Permissions, vérifiez que vous êtes le propriétaire du document. Seul le propriétaire d'un document peut le partager.
4. Choisissez Edit.
5. Pour partager la commande publiquement, sélectionnez Public, puis Save. Pour partager la commande de façon privée, sélectionnez Private, saisissez l'ID de compte AWS, puis sélectionnez Add permission et Save.

Partager un document (ligne de commande)

La procédure suivante nécessite la spécification d'une région AWS pour votre session de ligne de commande.

1. Ouvrez AWS CLI ou Outils AWS pour Windows PowerShell sur votre ordinateur local et exécutez la commande suivante pour spécifier vos informations d'identification.

   Linux

   ```
   aws config

   AWS Access Key ID: [your key]
   AWS Secret Access Key: [your key]
   Default region name: region
   Default output format [None]:
   ```

   Windows

   ```
   aws config

   AWS Access Key ID: [your key]
   AWS Secret Access Key: [your key]
   ```

```
Default region name: region
Default output format [None]:
```

PowerShell

```
Set-AWSCredentials -AccessKey your key -SecretKey your key
Set-DefaultAWSRegion -Region region
```

région représente l'identifiant d'une région AWS prise en charge par AWS Systems Manager, comme `us-east-2` pour la région Région USA Est (Ohio). Pour voir la liste des valeurs de *region* prises en charge, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

2. Utilisez la commande suivante pour répertorier tous les documents SSM qui sont disponibles. La liste inclut les documents que vous avez créés et ceux qui ont été partagés avec vous.

   Linux

   ```
   aws ssm list-documents
   ```

   Windows

   ```
   aws ssm list-documents
   ```

   PowerShell

   ```
   Get-SSMDocumentList
   ```

3. Utilisez la commande suivante pour obtenir un document spécifique.

   Linux

   ```
   aws ssm get-document \
       --name document name
   ```

   Windows

   ```
   aws ssm get-document ^
       --name document name
   ```

   PowerShell

   ```
   Get-SSMDocument `
       -Name document name
   ```

4. Utilisez la commande suivante pour obtenir la description du document.

   Linux

   ```
   aws ssm describe-document \
       --name document name
   ```

   Windows

   ```
   aws ssm describe-document ^
   ```

```
    --name document name
```

PowerShell

```
Get-SSMDocumentDescription `
    -Name document name
```

5. Utilisez la commande suivante pour afficher les autorisations définies pour le document.

   Linux

```
aws ssm describe-document-permission \
    --name document name \
    --permission-type Share
```

   Windows

```
aws ssm describe-document-permission ^
    --name document name ^
    --permission-type Share
```

   PowerShell

```
Get-SSMDocumentPermission `
    -Name document name `
    -PermissionType Share
```

6. Utilisez la commande suivante pour modifier les autorisations définies pour le document et partager celui-ci. Vous devez être le propriétaire du document pour modifier les autorisations. Cette commande partage le document de façon privée avec la personne spécifiée, en fonction de l'ID de compte AWS de cette personne.

   Linux

```
aws ssm modify-document-permission \
    --name document name \
    --permission-type Share \
    --account-ids-to-add AWS account ID
```

   Windows

```
aws ssm modify-document-permission ^
    --name document name ^
    --permission-type Share ^
    --account-ids-to-add AWS account ID
```

   PowerShell

```
Edit-SSMDocumentPermission `
    -Name document name `
    -PermissionType Share `
    -AccountIdsToAdd AWS account ID
```

7. Utilisez la commande suivante pour partager un document publiquement.

Linux

```
aws ssm modify-document-permission \
    --name document name \
    --permission-type Share \
    --account-ids-to-add 'all'
```

Windows

```
aws ssm modify-document-permission ^
    --name document name ^
    --permission-type Share ^
    --account-ids-to-add "all"
```

PowerShell

```
Edit-SSMDocumentPermission `
    -Name document name `
    -PermissionType Share `
    -AccountIdsToAdd ('all')
```

Modification des autorisations d'un document SSM partagé

Si vous partagez une commande, les utilisateurs peuvent afficher et utiliser cette commande jusqu'à ce que vous supprimiez l'accès au document SSM ou supprimiez le document SSM. Cependant, vous ne pouvez pas supprimer un document tant qu'il est partagé. Vous devez d'abord arrêter le partage, puis supprimer le document.

Rubriques

- Arrêt du partage d'un document (console) (p. 1280)
- Arrêter le partage d'un document (ligne de commande) (p. 1281)

Arrêt du partage d'un document (console)

Arrêt du partage d'un document

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Documents dans le panneau de navigation.
3. Dans la liste des documents, sélectionnez le document que vous souhaitez arrêter de partager, puis choisissez Afficher les détails. Sur l'onglet Permissions, vérifiez que vous êtes le propriétaire du document. Seul le propriétaire d'un document peut arrêter le partage de celui-ci.
4. Choisissez Edit.
5. Choisissez X pour supprimer l'ID de compte AWS qui ne doit plus avoir accès à la commande, puis sélectionnez Enregistrer.

Arrêter le partage d'un document (ligne de commande)

Ouvrez l'AWS CLI ou Outils AWS pour Windows PowerShell sur votre ordinateur local et exécutez la commande suivante pour arrêter le partage d'une commande.

Linux

```
aws ssm modify-document-permission \
    --name document name \
    --permission-type Share \
    --account-ids-to-remove 'AWS account ID'
```

Windows

```
aws ssm modify-document-permission ^
    --name document name ^
    --permission-type Share ^
    --account-ids-to-remove "AWS account ID"
```

PowerShell

```
Edit-SSMDocumentPermission `
    -Name document name `
    -PermissionType Share `
    -AccountIdsToRemove AWS account ID
```

Utilisation de documents SSM partagés

Lorsque vous partagez un SSM document, le système génère un Amazon Resource Name (ARN) et l'affecte à la commande. Si vous sélectionnez et exécutez un document partagé à partir de la console Amazon EC2, vous ne voyez pas l'ARN. Toutefois, si vous voulez exécuter un document SSM partagé à partir d'une application de ligne de commande, vous devez spécifier un ARN complet. L'ARN complet d'un document SSM s'affiche lorsque vous exécutez la commande permettant de répertorier les documents.

> **Note**
>
> Vous n'êtes pas obligé de spécifier ARNs pour les documents publics AWS (documents qui commencent par AWS-*) ou les documents que vous possédez.

Utiliser un document SSM partagé (ligne de commande)

Pour répertorier tous les SSM documents publics

Linux

```
aws ssm list-documents \
    --filters Key=Owner,Values=Public
```

Windows

```
aws ssm list-documents ^
    --filters Key=Owner,Values=Public
```

PowerShell

```
$filter = New-Object Amazon.SimpleSystemsManagement.Model.DocumentKeyValuesFilter
```

```
$filter.Key = "Owner"
$filter.Values = "Public"

Get-SSMDocumentList `
    -Filters @($filter)
```

Pour répertorier les SSM documents privés qui ont été partagés avec vous

Linux

```
aws ssm list-documents \
    --filters Key=Owner,Values=Private
```

Windows

```
aws ssm list-documents ^
    --filters Key=Owner,Values=Private
```

PowerShell

```
$filter = New-Object Amazon.SimpleSystemsManagement.Model.DocumentKeyValuesFilter
$filter.Key = "Owner"
$filter.Values = "Private"

Get-SSMDocumentList `
    -Filters @($filter)
```

Pour répertorier tous les SSM documents disponibles

Linux

```
aws ssm list-documents
```

Windows

```
aws ssm list-documents
```

PowerShell

```
Get-SSMDocumentList
```

Pour obtenir des informations sur un SSM document qui a été partagé avec vous

Linux

```
aws ssm describe-document \
    --name arn:aws:ssm:us-east-2:12345678912:document/documentName
```

Windows

```
aws ssm describe-document ^
    --name arn:aws:ssm:us-east-2:12345678912:document/documentName
```

PowerShell

```
Get-SSMDocumentDescription `
    -Name arn:aws:ssm:us-east-2:12345678912:document/documentName
```

Pour exécuter un SSM document partagé

Linux

```
aws ssm send-command \
    --document-name arn:aws:ssm:us-east-2:12345678912:document/documentName \
    --instance-ids ID
```

Windows

```
aws ssm send-command ^
    --document-name arn:aws:ssm:us-east-2:12345678912:document/documentName ^
    --instance-ids ID
```

PowerShell

```
Send-SSMCommand `
    -DocumentName arn:aws:ssm:us-east-2:12345678912:document/documentName `
    -InstanceIds ID
```

Recherche de SSM documents

Vous pouvez rechercher AWS Systems Manager des documents dans le magasin de SSM documents à l'aide d'une recherche de texte libre ou d'une recherche basée sur un filtre. Cette section décrit comment utiliser les deux fonctionnalités pour rechercher un SSM document .

Utilisation de la recherche en texte libre

La zone de recherche de la Systems Manager page Documents prend en charge la recherche en texte libre. La recherche en texte libre compare le terme recherché ou les termes que vous saisissez avec le nom du document dans chaque SSM document. Si vous entrez un seul terme à rechercher, par exemple **`ansible`** , Systems Manager renvoie tous les SSM documents dans lesquels ce terme a été découvert. Si vous entrez plusieurs termes recherchés, Systems Manager effectue une recherche à l'aide d'une `OR` instruction . Par exemple, si vous spécifiez **`ansible`** et **`linux`** , la recherche renvoie tous les documents dont le nom contient l'un ou l'autre mot-clé.

Si vous entrez un terme de recherche en texte libre et choisissez une option de recherche, telle que Platform type, la recherche utilise une `AND` instruction et renvoie tous les documents avec le mot-clé dans leur nom et le type de fichier platfrom spécifié.

> **Note**
>
> Notez les détails suivants sur la recherche en texte libre.
>
> - La recherche en texte libre n'est pas sensible à la casse.
> - Les termes à rechercher nécessitent un minimum de trois caractères et un maximum de 20 caractères.

- La recherche en texte libre accepte jusqu'à cinq termes de recherche.
- Si vous entrez un espace entre les termes recherchés, le système inclut l'espace lors de la recherche.
- Vous pouvez combiner la recherche de texte libre avec d'autres options de recherche telles que le type de document ou le type de plateforme.
- Le filtre Document Name Prefix et la recherche de texte libre ne peuvent pas être utilisés ensemble. Elles sont mutuellement exclusives.

Pour rechercher un SSM document

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Documents dans le panneau de navigation.
3. Saisissez vos termes de recherche dans la zone de recherche, puis appuyez sur Entrée.

Exécution d'une recherche de document en texte libre à l'aide de l'AWS CLI

Pour effectuer une recherche de document en texte libre à l'aide de l'interface de ligne de commande

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI.

   Pour de plus amples informations, veuillez consulter Installer ou mettre à niveau les outils de ligne de commande AWS (p. 65).
2. Pour effectuer une recherche de document en texte libre avec un seul terme, exécutez la commande suivante. Dans cette commande, remplacez *search_term* avec vos propres informations.

   ```
   aws ssm list-documents --filters Key="SearchKeyword",Values="search_term"
   ```

   Voici un exemple :

   ```
   aws ssm list-documents --filters Key="SearchKeyword",Values="aws-asg" --region us-
   east-2
   ```

   Pour effectuer une recherche à l'aide de plusieurs termes qui créent une `AND` instruction , exécutez la commande suivante. Dans cette commande, remplacez *search_term_1* et *search_term_2* avec vos propres informations.

   ```
   aws ssm list-documents --filters
    Key="SearchKeyword",Values="search_term_1","search_term_2","search_term_3" --region
    us-east-2
   ```

   Voici un exemple :

   ```
   aws ssm list-documents --filters Key="SearchKeyword",Values="aws-asg","aws-
   ec2","restart" --region us-east-2
   ```

Utilisation de filtres

La Systems Manager page Documents affiche automatiquement les filtres suivants lorsque vous choisissez la zone de recherche.

- Préfixe du nom du document
- Types de plateformes
- Type de document
- Clé de balise

Vous pouvez rechercher des SSM documents à l'aide d'un seul filtre. Si vous souhaitez renvoyer un ensemble de SSM documents plus spécifique, vous pouvez appliquer plusieurs filtres. Voici un exemple de recherche qui utilise les types de plateforme et les filtres de préfixe Nom du document.

Si vous appliquez plusieurs filtres, Systems Manager crée différentes instructions de recherche en fonction des filtres que vous choisissez :

- Si vous appliquez le même filtre plusieurs fois, par exemple le préfixe du nom du document, Systems Manager effectue une recherche à l'aide d'une `OR` instruction . Par exemple, si vous spécifiez un filtre de préfixe de nom de document = et un second filtre de préfixe de nom de **`AWS`** document =, la recherche renvoie tous les documents avec le préfixe « AWS » et tous les documents avec le préfixe « Lambda ».**`Lambda`**
- Si vous appliquez des filtres différents, par exemple le préfixe du nom du document et les types de plateforme, Systems Manager effectue une recherche à l'aide d'une `AND` instruction . Par exemple, si vous spécifiez un préfixe de nom de document= **`AWS`** filtre et un filtre de types de plateforme=, la recherche renvoie tous les documents avec le préfixe « AWS » qui sont spécifiques à la plateforme Linux.**`Linux`**

> **Note**
>
> Les recherches qui utilisent des filtres sont sensibles à la casse.

Exécution de documents Systems Manager de commande à partir d'emplacements distants

Vous pouvez exécuter Systems Manager des documents (documents SSM) à partir d'emplacements distants à l'aide du `AWS-RunDocument` document SSM prédéfini. Ce document prend actuellement en charge l'exécution des documents SSM stockés aux emplacements suivants :

- GitHubLes référentiels (publics et privés)
- Compartiments Amazon S3
- Systems Manager

Bien que vous puissiez également exécuter des documents distants à l'aide d' State Manager ou d'Automation, la procédure suivante décrit uniquement comment exécuter SSM des documents distants à l'aide d' Fonctionnalité Exécuter la commande dans la AWS Systems Manager console .

> Note
>
> `AWS-RunDocument`peut être utilisé pour exécuter uniquement des documents SSM de type commande, et non d'autres types tels que les documents Automation. utilise le `AWS-RunDocument` plug-in .`aws:downloadContent` Pour plus d'informations sur le `aws:downloadContent` plug-in , consultez aws:downloadContent (p. 1248) .

Avant de commencer

Avant d'exécuter un document distant, vous devez effectuer les tâches suivantes :

- Créez un document SSM de commande et enregistrez-le dans un emplacement distant. Pour plus d'informations, veuillez consulter Création de documents Systems Manager (p. 1267)
- Si vous prévoyez d'exécuter un document distant stocké dans un GitHub référentiel privé, vous devez créer un Systems Manager `SecureString` paramètre pour votre jeton d'accès GitHub de sécurité. Vous ne pouvez pas accéder à un document distant dans un GitHub référentiel privé en transmettant manuellement votre jeton via SSH. Le jeton d'accès doit être transmis en tant que paramètre Systems Manager `SecureString`. Pour plus d'informations sur la création d'un paramètre `SecureString`, consultez Création de paramètres Systems Manager (p. 243).

Exécuter un document distant (console)

Pour exécuter un document distant

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fonctionnalité Exécuter la commande.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Fonctionnalité Exécuter la commande.
3. Choisissez Exécuter une commande.
4. Dans la liste Document, choisissez AWS-RunDocument.
5. Dans Paramètres de commande, pour Type de source, choisissez une option.

   - Si vous choisissez GitHub , spécifiez les informations d'information source au format suivant :

```
{
```

```
    "owner": "owner_name",
    "repository": "repository_name",
    "path": "path_to_document",
    "getOptions":"branch:branch_name",
    "tokenInfo": "{{ssm-secure:secure-string-token}}"
}
```

Exemple :

```
{
    "owner":"TestUser",
    "repository":"GitHubTestExamples",
    "path":"scripts/python/test-script",
    "getOptions":"branch:exampleBranch",
    "tokenInfo":"{{ssm-secure:my-secure-string-token}}"
}
```

Note

`getOptions`Les sont des options supplémentaires permettant de récupérer du contenu à partir d'une branche autre que master ou d'une validation spécifique dans le référentiel. `getOptions`peut être omis si vous utilisez la dernière validation dans la branche principale. Le `branch` paramètre est obligatoire uniquement si votre document SSM est stocké dans une branche autre que `master` .
Pour utiliser la version de votre document SSM dans une validation particulière dans votre référentiel, utilisez `commitID` avec `getOptions` au lieu d' `branch` . Par exemple :

```
"getOptions": "commitID:bbc1ddb94...b76d3bEXAMPLE",
```

- Si vous choisissez S3, spécifiez les informations source au format suivant :

```
{"path":"URL_to_document_in_S3"}
```

Exemple :

```
{"path":"https://s3.amazonaws.com/DOC-EXAMPLE-BUCKET/scripts/ruby/mySSMdoc.json"}
```

- Si vous choisissez SSMDocument , spécifiez les informations d'informations source au format suivant :

```
{"name": "document_name"}
```

Exemple :

```
{"name": "mySSMdoc"}
```

6. Dans le champ Paramètres du document, entrez les paramètres du document SSM distant. Par exemple, si vous exécutez le document AWS-RunPowerShell, vous pouvez spécifier :

```
{"commands": ["date", "echo \"Hello World\""]}
```

Si vous exécutez le document AWS-ConfigureAWSPack, vous pouvez spécifier :

```
{
   "action":"Install",
   "name":"AWSPVDriver"
```

```
}
```

7. Dans la section Cibles, identifiez les instances sur lesquelles vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant manuellement les instances ou en indiquant un groupe de ressources.

   Note

   Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.

8. Pour Autres paramètres :

   - Pour Comment (Commentaire), saisissez des informations à propos de cette commande.
   - Pour Délai (secondes), précisez le nombre de secondes durant lesquelles le système doit attendre avant de mettre en échec l'exécution de la commande globale.

9. (Facultatif) Pour Contrôle du débit :

   - Pour Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage d'instances sur lesquelles exécuter simultanément la commande.

     Note

     Si vous avez sélectionné des cibles en indiquant des balises appliquées à des instances gérées ou en indiquant des groupes de ressources AWS, et que vous n'êtes pas sûr du nombre d'instances ciblées, limitez le nombre d'instances pouvant exécuter simultanément le document en spécifiant un pourcentage.

   - Pour Seuil d'erreur, spécifiez quand arrêter l'exécution de la commande sur d'autres instances après son échec sur un nombre ou un pourcentage d'instances. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les instances sur lesquelles la commande est en cours de traitement peuvent également envoyer des erreurs.

10. (Facultatif) Pour Output options (Options de sortie), pour enregistrer la sortie de la commande dans un fichier, cochez la case Write command output to an S3 bucket (Écrire la sortie de commande vers un compartiment S3). Entrez les noms de compartiment et de préfixe (dossier) dans les zones.

    Note

    Les autorisations S3 qui accordent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour de plus amples informations, veuillez consulter Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). En outre, si le compartiment S3 spécifié se trouve dans un compte AWS différent, assurez-vous que le profil d'instance associé à l'instance dispose des autorisations nécessaires pour écrire dans ce compartiment.

11. Dans la section SNS notifications (Notifications SNS), si vous souhaitez que des notifications sur le statut de l'exécution de la commande soient envoyées, cochez la case Enable SNS notifications (Activer les notifications SNS).

    Pour plus d'informations sur la configuration des notifications Amazon SNS pour la fonctionnalité Exécuter la commande, consultez Surveillance des changements de statut Systems Manager à l'aide de notifications Amazon SNS (p. 1344).

12. Cliquez sur Exécuter.

Note

Pour de plus amples informations sur le redémarrage des serveurs et des instances en appelant des scripts à l'aide de Fonctionnalité Exécuter la commande, veuillez consulter Redémarrage d'une instance gérée à partir de scripts (p. 963).

Sécurité dans le AWS Systems Manager

Chez AWS, la sécurité dans le cloud est notre priorité numéro 1. En tant que client AWS, vous bénéficiez d'un centre de données et d'une architecture réseau conçus pour répondre aux exigences des organisations les plus pointilleuses en termes de sécurité.

La sécurité est une responsabilité partagée entre AWS et vous-même. Le modèle de responsabilité partagée décrit ceci comme la sécurité du cloud et la sécurité dans le cloud :

- La sécurité du cloud – AWS est responsable de la protection de l'infrastructure qui exécute des services AWS dans le cloud AWS. AWS vous fournit également les 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 conformité AWS. Pour en savoir plus sur les programmes de conformité qui s'appliquent à AWS Systems Manager , consultez Services AWS concernés par le programme de conformité – Services .
- Sécurité dans le cloud – Votre responsabilité est déterminée par le service AWS que vous utilisez. Vous êtes également responsable d'autres facteurs, y compris la sensibilité de vos données, les exigences de votre entreprise,et la législation et la réglementation applicables.

Cette documentation vous aide à comprendre comment appliquer le modèle de responsabilité partagée lors de l'utilisation de Systems Manager. Les rubriques suivantes montrent comment configurer Systems Manager pour répondre à vos objectifs de sécurité et de conformité. Vous pouvez également apprendre à utiliser d'autres services AWS capables de vous aider à surveiller et à sécuriser vos ressources Systems Manager.

Rubriques

- Protection des données dans le AWS Systems Manager (p. 1290)
- Identity and Access Management pour AWS Systems Manager (p. 1293)
- Utilisation des rôles liés à un service pour Systems Manager (p. 1312)
- Journalisation et surveillance dans AWS Systems Manager (p. 1318)
- Validation de la conformité pour le AWS Systems Manager (p. 1320)
- Résilience dans AWS Systems Manager (p. 1321)
- Sécurité de l'infrastructure dans AWS Systems Manager (p. 1321)
- Configuration et analyse des vulnérabilités dans AWS Systems Manager (p. 1321)
- Bonnes pratiques de sécurité pour Systems Manager (p. 1321)

Protection des données dans le AWS Systems Manager

Le AWS Systems Manager se conforme au modèle de responsabilité partagée AWS, qui inclut des directives et des régulations pour la protection des données. AWS est responsable de la protection de l'infrastructure globale qui exécute tous les services AWS. AWS conserve le contrôle des données hébergées sur cette infrastructure, y compris les contrôles de configuration de sécurité pour traiter le

contenu et les données personnelles des clients. Les clients AWS et les partenaires d'APN, qu'ils agissent en tant que contrôleurs ou responsables du traitement des données, sont responsables de toutes les données personnelles qu'ils mettent sur le cloud AWS.

La protection des données fait référence au fait de protéger les données en transit (lorsqu'elles sont transmises à ou à partir de Systems Manager) et au repos (lorsqu'elles sont stockées dans les centres de données AWS).

Pour des raisons de protection des données, nous vous recommandons de protéger vos identifiants de compte AWS et de configurer des comptes utilisateurs individuels avec AWS Identity and Access Management (IAM), afin que chaque utilisateur reçoive uniquement les permissions nécessaires pour accomplir ses tâches. Nous vous recommandons également de sécuriser vos données de la manière suivante :

- Utilisation de l'authentification multi-facteurs (MFA) avec chaque compte.
- Utilisez SSL/TLS pour communiquer avec des ressources AWS. Nous recommandons TLS 1.2 ou version ultérieure.
- Configurez l'API et la journalisation des activités utilisateur avec AWS CloudTrail.
- Utilisez des solutions de chiffrement AWS, ainsi que tous les contrôles de sécurité par défaut au sein des services AWS.
- Utilisez des services de sécurité gérés avancés comme Amazon Macie, qui contribuent à la découverte et à la sécurisation des données personnelles stockées dans Amazon S3.
- Si vous avez besoin de modules cryptographiques validés FIPS 140-2 lors de l'accès à 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 la norme FIPS 140-2.

Nous vous recommandons vivement de ne jamais placer d'informations identifiables sensibles, telles que des numéros de compte de vos clients, dans des champs de formulaire comme le Nom. Cela vaut également lorsque vous utilisez Systems Manager ou d'autres AWS services à l'aide de la console , de l'API AWS CLI , de l' ou AWS SDKs de l' . Toutes les données que vous entrez dans Systems Manager ou d'autres services peuvent être récupérées pour insertion dans des journaux de diagnostic. Lorsque vous fournissez une URL à un serveur externe, n'incluez pas les informations d'identification non chiffrées dans l'URL pour valider votre demande adressée au serveur.

Pour de plus amples informations sur la protection des données, veuillez consulter le billet de blog Modèle de responsabilité partagée AWS et RGPD sur le Blog sur la sécurité d'AWS.

Chiffrement des données

Chiffrement au repos

Paramètres Parameter Store

Les types de paramètres que vous pouvez créer dans Parameter Store incluent `String`, `StringList` et `SecureString`.

Pour chiffrer les valeurs de paramètre `SecureString`, Parameter Store utilise une clé principale client (CMK) AWS Key Management Service (AWS KMS). AWS KMS utilise soit une clé CMK gérée par le client, soit une clé CMK gérée par AWS pour chiffrer la valeur du paramètre dans une base de données gérée par AWS.

> **Important**
>
> Ne stockez pas de données sensibles dans un paramètre `StringList` ou `String`. Pour toutes les données sensibles qui doivent rester chiffrées, utilisez uniquement le type de paramètre `SecureString`.

Pour de plus amples informations, veuillez consulter Qu'est-ce qu'un paramètre ? (p. 220) et Restriction de l'accès aux paramètres Systems Manager à l'aide des stratégies IAM (p. 223).

Contenu des compartiments Amazon S3

Dans le cadre de vos opérations Systems Manager, vous pouvez choisir de charger ou de stocker des données dans un ou plusieurs compartiments Amazon Simple Storage Service (Amazon S3).

Pour plus d'informations sur le chiffrement des compartiments S3, consultez Protection des données à l'aide d'un chiffrement et Protection des données dans Amazon S3 dans le Amazon Simple Storage Service Manuel du développeur.

Voici les types de données que vous pouvez charger ou stocker dans les compartiments S3 dans le cadre de vos activités Systems Manager.

- Sortie des commandes Fonctionnalité Exécuter la commande
- Packages Distributor
- Gestionnaire de correctifsJournaux d'opérations d'application de correctifs
- listes de remplacement des Gestionnaire de correctifs correctifs
- Scripts ou Playbooks Ansible à exécuter dans un flux de travail de document Automation
- ProfilInSpecs Chef à utiliser avec les analyses de conformité
- Journaux de l'AWS CloudTrail
- Journaux d'historique de session Session Manager
- Rapports de l'Explorateur ou OpsData deOpsCenter
- Modèles AWS CloudFormation à utiliser avec les flux de travail Automation
- Données de conformité issues d'une analyse de synchronisation de données de ressources
- Sortie des demandes de création ou de modification de l'association State Manager sur les instances gérées
- Documents SSM personnalisés que vous pouvez exécuter à l'aide du document `AWS-RunDocument` géré par AWS

Groupes de journaux CloudWatch Logs

Dans le cadre de vos opérations Systems Manager, vous pouvez choisir de transmettre en continu des données à un ou plusieurs groupes de journaux Amazon CloudWatch Logs.

Pour plus d'informations sur le chiffrement des groupes de journaux CloudWatch Logs, consultez Chiffrement des données de journal dans CloudWatch Logs dans le Amazon CloudWatch Logs User Guide.

Voici les types de données que vous avez pu transmettre en continu à un groupe de journaux CloudWatch Logs dans le cadre de vos activités Systems Manager.

- Sortie des commandes Fonctionnalité Exécuter la commande
- Sortie des scripts exécutés à l'aide de l'action `aws:executeScript` dans un document Automation
- Journaux d'historique de session Session Manager
- Journaux de l'agent SSM sur vos instances gérées

Chiffrement en transit

Nous vous recommandons d'utiliser un protocole de chiffrement tel que Transport Layer Security (TLS) pour chiffrer les données sensibles en transit entre les clients et vos instances.

Systems Manager fournit la prise en charge suivante pour le chiffrement de vos données en transit.

Connexions aux points de terminaison d'API Systems Manager

Systems Manager Les points de terminaison d'API ne prennent en charge que des connexions sécurisées sur HTTPS. Lorsque vous gérez des ressources Systems Manager avec AWS Management Console, le kit SDK AWS ou l'API Systems Manager, toutes les communications sont chiffrées à l'aide du protocole TLS (Transport Layer Security). Pour obtenir la liste complète des points de terminaison d'API, consultez Régions AWS et points de terminaison dans le Référence générale d'Amazon Web Services.

Instances gérées

AWS assure une connectivité sûre et privée entre des instances EC2. En outre, nous chiffrons automatiquement le trafic en transit entre les instances prises en charge dans le même VPC ou dans un appairage VPCs , à l'aide d'algorithmes AEAD avec chiffrement 256 bits. Cette fonction de chiffrement utilise les capacités de déchargement du matériel sous-jacent, sans impact sur la performance de réseau. Les instances prises en charge sont : C5n, G4, I3en, M5dn, M5n, P3dn, R5dn et R5n.

Sessions Session Manager

Par défaut, Session Manager utilise TLS 1.2 pour chiffrer les données de session transmises entre les machines locales des utilisateurs de votre compte et vos instances EC2. Vous pouvez également choisir de chiffrer davantage les données en transit à l'aide d'une clé principale client (CMK) créée dans AWS Key Management Service.

Accès via Fonctionnalité Exécuter la commande

Par défaut, l'accès distant à vos instances via Fonctionnalité Exécuter la commande est chiffré à l'aide de TLS 1.2 et les demandes de création d'une connexion sont chiffrées à l'aide d' SigV4 .

Confidentialité du trafic inter-réseaux

Vous pouvez utiliser Amazon Virtual Private Cloud (Amazon VPC) pour créer des frontières entre les ressources de vos instances gérées et contrôler le trafic entre celles-ci, votre réseau sur site et Internet. Pour de plus amples informations, veuillez consulter (Facultatif) Créer un point de terminaison de VPC (p. 42).

Pour de plus amples informations sur la sécurité Amazon Virtual Private Cloud, veuillez consulter Sécurité dans le Amazon VPC Guide de l'utilisateur.

Identity and Access Management pour AWS Systems Manager

AWS Identity and Access Management (IAM) est un service AWS qui permet à un administrateur de contrôler en toute sécurité l'accès aux ressources AWS. Les administrateurs IAM contrôlent qui peut être authentifié (connecté) et autorisé (disposant des autorisations) à utiliser les ressources Systems Manager. IAM est un service AWS que vous pouvez utiliser sans frais supplémentaires.

Rubriques

- Audience (p. 1294)
- Authentification avec des identités (p. 1294)
- Gestion de l'accès à l'aide de stratégies (p. 1296)

- Fonctionnement d'AWS Systems Manager avec IAM (p. 1298)
- Exemples de stratégies basées sur l'identité AWS Systems Manager (p. 1304)
- Résolution des problèmes d'identité et d'accès AWS Systems Manager (p. 1310)

Audience

Votre utilisation d'AWS Identity and Access Management (IAM) évolue selon la tâche que vous réalisez dans Systems Manager.

Utilisateur du service – Si vous utilisez le service Systems Manager pour effectuer votre tâche, votre administrateur vous fournit les informations d'identification et les autorisations dont vous avez besoin. Plus vous utiliserez de fonctionnalités Systems Manager pour effectuer votre travail, plus vous pourrez avoir besoin d'autorisations supplémentaires. Comprendre la gestion des accès peut vous aider à demander à votre administrateur les autorisations appropriées. Si vous ne pouvez pas accéder à une fonctionnalité dans Systems Manager, consultez Résolution des problèmes d'identité et d'accès AWS Systems Manager (p. 1310).

Administrateur du service –Si vous êtes le responsable des ressources Systems Manager de votre entreprise, vous bénéficiez probablement d'un accès total à Systems Manager. C'est à vous de déterminer les fonctionnalités et les ressources Systems Manager auxquelles vos employés pourront accéder. Vous devez ensuite soumettre les demandes à votre administrateur IAM pour modifier les autorisations des utilisateurs de votre service. Consultez les informations sur cette page pour comprendre les concepts de base d'IAM. Pour en savoir plus sur la façon dont votre entreprise peut utiliser IAM avec Systems Manager, consultez Fonctionnement d'AWS Systems Manager avec IAM (p. 1298).

Administrateur IAM – Si vous êtes un administrateur IAM, vous souhaiterez peut-être obtenir des détails sur la façon dont vous pouvez écrire des stratégies pour gérer l'accès à Systems Manager. Pour obtenir des exemples de stratégies Systems Manager basées sur l'identité que vous pouvez utiliser dans IAM, consultez Exemples de stratégies basées sur l'identité AWS Systems Manager (p. 1304).

Authentification avec des identités

L'authentification correspond au processus par lequel vous vous connectez à AWS via vos informations d'identification. Pour plus d'informations sur la signature à l'aide d'AWS Management Console, consultez La console et la page de connexion IAM dans le IAM Guide de l'utilisateur.

Vous devez être authentifié (connecté à AWS) en tant que Utilisateur racine d'un compte AWS ou utilisateurIAM, ou en assumant un rôle IAM. Vous pouvez également utiliser l'authentification de connexion unique de votre entreprise ou vous connecter via Google ou Facebook. Dans ce cas, votre administrateur aura précédemment configuré une fédération d'identités avec des rôles IAM. Lorsque vous accédez à AWS avec des informations d'identification d'une autre entreprise, vous assumez indirectement un rôle.

Pour vous connecter directement à la AWS Management Console, utilisez votre mot de passe avec votre e-mail utilisateur racine ou votre nom d'utilisateur IAM. Vous pouvez accéder à AWS par programmation avec vos clés d'accès utilisateur utilisateur racine ou IAM. AWS fournit un kit de développement logiciel (SDK) et des outils de ligne de commande pour signer de manière cryptographique votre requête avec vos informations d'identification. Si vous n'utilisez pas les outils AWS, vous devez signer la demande vous-même. Pour ce faire, utilisez Signature Version 4, un protocole permettant d'authentifier les demandes d'API entrantes. Pour en savoir plus sur l'authentification des demandes, consultez Processus de signature Signature Version 4 dans la documentation AWS General Reference.

Quelle que soit la méthode d'authentification que vous utilisez, vous devrez peut-être également fournir des informations de sécurité supplémentaires. Par exemple, AWS vous recommande d'utiliser l'authentification multi-facteurs (MFA) pour améliorer la sécurité de votre compte. Pour en savoir plus, consultez Utilisation de Multi-Factor Authentication (MFA) dans AWS dans le IAM Guide de l'utilisateur.

Utilisateur racine d'un compte AWS

Lorsque vous créez un compte AWS, vous commencez avec une seule identité de connexion disposant d'un accès complet à tous les services et ressources AWS du compte. Cette identité est appelée la utilisateur racinedu compte AWS et elle est accessible après connexion à l'aide de l'adresse e-mail et du mot de passe utilisés pour la création du compte. Il est vivement recommandé de ne pas utiliser l'utilisateur racine pour vos tâches quotidiennes, y compris pour les tâches administratives. Au lieu de cela, respectez la bonne pratique qui consiste à avoir recours à l'utilisateur racine uniquement pour créer le premier utilisateurIAM. Ensuite, mettez en sécurité les informations d'identification de l'utilisateur racine et utilisez-les pour effectuer uniquement certaines tâches de gestion des comptes et des services.

Utilisateurs et groupes IAM

Un utilisateur IAM est une identité dans votre compte AWS qui dispose d'autorisations spécifiques pour une seule personne ou application. Un utilisateur IAM peut disposer d'informations d'identification à long terme comme un nom d'utilisateur et un mot de passe ou un ensemble de clés d'accès. Pour savoir comment générer des clés d'accès, consultez Gestion des clés d'accès pour les utilisateurs IAM dans le IAM Guide de l'utilisateur. Lorsque vous générez des clés d'accès pour un utilisateur IAM, veillez à afficher et enregistrer la paire de clés de manière sécurisée. Vous ne pourrez plus récupérer la clé d'accès secrète à l'avenir. Au lieu de cela, vous devrez générer une nouvelle paire de clés d'accès.

Un groupe IAM est une identité qui spécifie un ensemble d'utilisateurs IAM. Vous ne pouvez pas vous connecter en tant que groupe. Vous pouvez utiliser les groupes pour spécifier des autorisations pour plusieurs utilisateurs à la fois. Les groupes permettent de gérer plus facilement les autorisations pour de grands ensembles d'utilisateurs. Par exemple, vous pouvez avoir un groupe nommé Admins IAM et accorder à ce groupe les autorisations leur permettant d'administrer des ressources IAM.

Les utilisateurs sont différents des rôles. Un utilisateur est associé de manière unique à une personne ou une application, alors qu'un rôle est conçu pour être endossé par tout utilisateur qui en a besoin. Les utilisateurs disposent d'informations d'identification permanentes, mais les rôles fournissent des informations d'identification temporaires. Pour plus de détails, consultez Quand créer un utilisateur IAM (au lieu d'un rôle) dans le IAM Guide de l'utilisateur.

Rôles IAM

Un rôle IAM est une entité au sein de votre compte AWS qui dispose d'autorisations spécifiques. Le concept ressemble à celui d'utilisateur IAM, mais un rôle n'est pas associé à une personne en particulier. Vous pouvez temporairement endosser un rôle IAM dans l'AWS Management Console grâce au changement de rôle. Vous pouvez obtenir un rôle en appelant une opération d'API AWS CLI ou AWS à l'aide d'une URL personnalisée. Pour plus d'informations sur les méthodes d'utilisation des rôles, consultez Utilisation de rôles IAM dans le IAM Guide de l'utilisateur.

Les rôles IAM avec des informations d'identification temporaires sont utiles dans les cas suivants :

- Autorisations utilisateur IAM temporaires – Un utilisateur IAM peut endosser un rôle IAM pour accepter différentes autorisations temporaires concernant une tâche spécifique.
- Accès d'utilisateurs fédérés – Au lieu de créer un utilisateur IAM, vous pouvez utiliser des identités d'utilisateur préexistantes provenant d'AWS Directory Service, de l'annuaire d'utilisateurs de votre entreprise ou d'un fournisseur d'identité web. On parle alors d'utilisateurs fédérés. AWS attribue un rôle à un utilisateur fédéré lorsque l'accès est demandé via un fournisseur d'identité. Pour plus d'informations sur les utilisateurs fédérés, consultez Utilisateurs fédérés et rôles dans le IAM Guide de l'utilisateur.
- Accès entre comptes – Vous pouvez utiliser un rôle IAM pour permettre à un utilisateur (mandataire de confiance) d'un compte différent d'accéder aux ressources de votre compte. Les rôles constituent le principal moyen d'accorder l'accès entre plusieurs comptes. Toutefois, certains services AWS vous permettent d'attacher une stratégie directement à une ressource (au lieu d'utiliser un rôle en tant que proxy). Pour en savoir plus sur la différence entre les rôles et les stratégies basées sur les ressources

pour l'accès entre comptes, consultez Différence entre les rôles IAM et les stratégies basées sur les ressources dans le IAM Guide de l'utilisateur.

- Accès à un service AWS –Un rôle de service est un rôle IAM qu'un service assume pour effectuer des actions dans votre compte en votre nom. Lorsque vous configurez certains environnements de services AWS, vous devez définir un rôle que ce service devra assumer. Ce rôle de service doit comprendre toutes les autorisations nécessaires pour que le service puisse accéder aux ressources AWS dont il a besoin. Les rôles de service varient d'un service à un service, mais nombre d'entre eux vous permettent de choisir vos autorisations, tant que vous respectez les exigences documentées pour le service en question. Les rôles de service fournissent un accès uniquement au sein de votre compte et ne peuvent pas être utilisés pour accorder l'accès à des services dans d'autres comptes. Vous créez, modifiez et supprimez un rôle de service à partir d'IAM. Par exemple, vous pouvez créer un rôle qui permet à Amazon Redshift d'accéder à un compartiment Amazon S3 en votre nom, puis de charger les données stockées dans ce compartiment dans un cluster Amazon Redshift. Pour plus d'informations, consultez Création d'un rôle pour déléguer des autorisations à un service AWS dans le IAM Guide de l'utilisateur.
- Applications qui s'exécutent sur Amazon EC2 –Vous pouvez utiliser un rôle IAM pour gérer des informations d'identification temporaires pour les applications qui s'exécutent sur une instance EC2 et effectuent des demandes d'API AWS CLI ou AWS. Cette solution est préférable au stockage des clés d'accès au sein de l'instance EC2. Pour attribuer un rôle AWS à une instance EC2 et le rendre disponible à toutes les applications associées, vous pouvez créer un profil d'instance attaché à l'instance. Un profil d'instance contient le rôle et permet aux programmes qui s'exécutent sur l'instance EC2 d'obtenir des informations d'identification temporaires. Pour plus d'informations, consultez Utilisation d'un rôle IAM pour accorder des autorisations à des applications s'exécutant sur des instances Amazon EC2 dans le IAM Guide de l'utilisateur.

Pour savoir si vous devez utiliser ces rôles IAM ou non, consultez Quand créer un rôle IAM (au lieu d'un utilisateur) dans le IAM Guide de l'utilisateur.

Gestion de l'accès à l'aide de stratégies

Vous contrôlez les accès dans AWS en créant des stratégies et en les attachant à des identités IAM ou à des ressources AWS. Une stratégie est un objet dans AWS qui, lorsqu'il est associé à une identité ou à une ressource, définit les autorisations de ces dernières. AWS évalue ces stratégies lorsqu'une entité (utilisateur racine, utilisateur IAM ou rôle IAM) envoie une demande. Les autorisations dans les stratégies déterminent si la demande est autorisée ou refusée. La plupart des stratégies sont stockées dans AWS en tant que documents JSON. Pour plus d'informations sur la structure et le contenu du document de stratégie JSON, consultez Présentation des stratégies JSON dans le IAM Guide de l'utilisateur.

Les stratégies permettent à un administrateur IAM de spécifier qui a accès aux ressources AWSet quelles actions ces personnes peuvent exécuter sur ces ressources. Chaque entité IAM (utilisateur ou rôle) démarre sans autorisation. En d'autres termes, par défaut, les utilisateurs ne peuvent rien faire, pas même changer leurs propres mots de passe. Pour autoriser un utilisateur à effectuer une opération, un administrateur doit associer une stratégie d'autorisations à ce dernier. Il peut également ajouter l'utilisateur à un groupe disposant des autorisations prévues. Lorsqu'un administrateur accorde des autorisations à un groupe, tous les utilisateurs de ce groupe se voient octroyer ces autorisations.

Les stratégies IAM définissent les autorisations d'une action quelle que soit la méthode que vous utilisez pour exécuter l'opération. Par exemple, supposons que vous disposiez d'une stratégie qui autorise l'action `iam:GetRole`. Un utilisateur avec cette stratégie peut obtenir des informations utilisateur à partir de l'AWS Management Console, de l'AWS CLI ou de l'API AWS.

Stratégies basées sur l'identité

Les stratégies basées sur l'identité sont des documents de stratégie d'autorisations JSON que vous pouvez attacher à une identité telle qu'un utilisateur, un rôle ou un groupe IAM. Ces stratégies contrôlent les actions que peut exécuter cette identité, sur quelles ressources et dans quelles conditions. Pour découvrir

comment créer une stratégie basée sur l'identité, consultez Création de stratégies IAM dans le IAM Guide de l'utilisateur.

Les stratégies basées sur l'identité peuvent être classées comme étant des stratégies en ligne ou des stratégies gérées. Les stratégies en ligne sont intégrées directement à un utilisateur, groupe ou rôle. Les stratégies gérées sont des stratégies autonomes que vous pouvez lier à plusieurs utilisateurs, groupes et rôles de votre compte AWS. Les stratégies gérées incluent les stratégies gérées par AWS et les stratégies gérées par le client. Pour découvrir comment choisir entre une politique gérée ou une politique en ligne, consultez Choix entre les stratégies gérées et les stratégies en ligne dans le IAM Guide de l'utilisateur.

Pour plus d'informations sur les stratégies gérées AWS pour Systems Manager, consultez Stratégies gérées par AWS pour AWS Systems Manager (p. 1303).

Stratégies basées sur les ressources

Les stratégies basées sur les ressources sont des documents de stratégie JSON que vous attachez à une ressource, telle qu'un compartiment Amazon S3. Les administrateurs de service peuvent utiliser ces stratégies pour définir les actions qu'un principal (membre de compte, utilisateur ou rôle) spécifié peut effectuer sur cette ressource et dans quelles conditions. Les stratégies basées sur les ressources sont des stratégies en ligne. Il ne s'agit pas de stratégies gérées basées sur les ressources.

Listes de contrôle d'accès (ACL)

Les listes de contrôle d'accès (ACL) constituent un type de stratégie permettant de définir les mandataires (membres de compte, utilisateurs ou rôles) ayant l'autorisation d'accéder à une ressource. Les listes de contrôle d'accès sont semblables aux stratégies basées sur les ressources, bien qu'elles n'utilisent pas le format de document de stratégie JSON. Amazon S3, AWS WAF et Amazon VPC sont des exemples de services prenant en charge les listes de contrôle d'accès. Pour en savoir plus sur les listes de contrôle d'accès, consultez Présentation de la liste de contrôle d'accès (ACL) dans le Manuel du développeur Amazon Simple Storage Service.

Autres types de stratégie

AWS prend en charge d'autres types de stratégies moins courantes. Ces types de stratégies peuvent définir le nombre maximal d'autorisations qui vous sont accordées par des types de stratégies plus courants.

- Limite d'autorisations – Une limite d'autorisations est une fonctionnalité avancée dans laquelle vous définissez les autorisations maximales qu'une stratégie basée sur l'identité peut accorder à une entité IAM (utilisateur ou rôle IAM). Vous pouvez définir une limite d'autorisations pour une entité. Les autorisations obtenues représentent la combinaison des stratégies basées sur l'identité de l'entité et de ses limites d'autorisations. Les stratégies basées sur les ressources qui spécifient l'utilisateur ou le rôle dans le champ `Principal` ne sont pas limitées par les limites d'autorisations. Un refus explicite dans l'une de ces stratégies remplace l'autorisation. Pour plus d'informations sur les limites d'autorisations, consultez Limites d'autorisations pour des entités IAM dans le IAM Guide de l'utilisateur.
- Stratégies de contrôle de service (SCP) – Les SCP sont des stratégies JSON qui spécifient le nombre maximal d'autorisations pour une organisation ou une unité d'organisation (OU) dans AWS Organizations. AWS Organizations est un service qui vous permet de regrouper et de gérer de façon centralisée plusieurs comptes AWS détenus par votre entreprise. Si vous activez toutes les fonctions d'une organisation, vous pouvez appliquer les stratégies de contrôle de service (SCP) à l'un ou à l'ensemble de vos comptes. La SCP limite les autorisations pour les entités dans les comptes membres, y compris dans chaque Utilisateur racine d'un compte AWS. Pour plus d'informations sur les Organisations et les SCP, consultez Fonctionnement des stratégies de contrôle de service dans le Manuel de l'utilisateur AWS Organizations.
- Stratégies de session – Les stratégies de session sont des stratégies avancées que vous transmettez en tant que paramètre lorsque vous créez par programmation une session temporaire pour un rôle

ou un utilisateur fédéré. Les autorisations de la session obtenue sont une combinaison des stratégies basées sur l'identité de l'utilisateur ou du rôle et des stratégies de session. Les autorisations peuvent également provenir d'une stratégie basée sur les ressources. Un refus explicite dans l'une de ces stratégies remplace l'autorisation. Pour de plus amples informations, veuillez consulter Stratégies de session dans le IAM Guide de l'utilisateur.

Plusieurs types de stratégie

Lorsque plusieurs types de stratégies s'appliquent à la requête, les autorisations obtenues sont plus compliquées à comprendre. Pour découvrir la façon dont AWS détermine s'il convient d'autoriser une demande en présence de plusieurs types de stratégies, consultez Logique d'évaluation de stratégies dans le IAM Guide de l'utilisateur.

Fonctionnement d'AWS Systems Manager avec IAM

Avant d'utiliser IAM pour gérer l'accès à Systems Manager, vous devez comprendre quelles fonctions IAM sont disponibles et peuvent être utilisées avec Systems Manager. Pour obtenir une vue d'ensemble de la façon dont Systems Manager et d'autres services AWS fonctionnent avec IAM, veuillez consulter Services AWS qui fonctionnent avec IAM dans le IAM Guide de l'utilisateur.

Rubriques

- Stratégies Systems Manager basées sur l'identité (p. 1298)
- Stratégies Systems Manager basées sur les ressources (p. 1302)
- Autorisation basée sur les balises Systems Manager (p. 1302)
- Rôles IAM Systems Manager (p. 1302)

Stratégies Systems Manager basées sur l'identité

Avec les stratégies basées sur une identité IAM, vous pouvez spécifier des actions et ressources autorisées ou refusées, ainsi que les conditions dans lesquelles les actions sont autorisées ou refusées. Systems Manager prend en charge des actions, ressources et clés de condition spécifiques. Pour en savoir plus sur tous les éléments que vous utilisez dans une stratégie JSON, consultez Références des éléments de stratégie JSON IAM dans le IAM Guide de l'utilisateur.

Actions

L'élément `Action` d'une stratégie basée sur une identité IAM décrit les actions spécifiques qui seront autorisées ou refusées par la stratégie. Les actions de stratégie possèdent généralement le même nom que l'opération d'API AWS associée. L'action est utilisée dans une stratégie pour permettre d'effectuer l'opération associée.

Les actions de stratégie dans Systems Manager utilisent le préfixe suivant avant l'action : `ssm:`. Par exemple, pour accorder à une personne l'autorisation de créer un Systems Manager paramètre avec l'opération Systems Manager `PutParameter` d'API, vous incluez l'`ssm:PutParameter`action dans sa stratégie. Les déclarations de stratégie doivent inclure un élément `Action` ou `NotAction`. Systems Manager définit son propre ensemble d'actions qui décrivent les tâches que vous pouvez effectuer avec ce service.

Pour spécifier plusieurs actions dans une seule déclaration, séparez-les par des virgules comme suit :

```
"Action": [
      "ssm:action1",
      "ssm:action2"
```

Vous pouvez aussi spécifier plusieurs actions à l'aide de caractères génériques (*). Par exemple, pour spécifier toutes les actions qui commencent par le mot `Describe`, incluez l'action suivante :

```
"Action": "ssm:Describe*"
```

Pour afficher la liste des actions Systems Manager, veuillez consulter Actions définies par AWS Systems Manager dans le IAM Guide de l'utilisateur.

Resources

L'élément `Resource` spécifie les objets auxquels l'action s'applique. Les instructions doivent inclure un élément `Resource` ou `NotResource`. Vous spécifiez une ressource à l'aide d'un ARN ou du caractère générique (*) pour indiquer que l'instruction s'applique à toutes les ressources.

Par exemple, la ressource de fenêtre de maintenance Systems Manager a le format ARN suivant.

```
arn:aws:ssm:region:account-id:maintenancewindow/window-id
```

Pour spécifier les fenêtres de maintenance mw-0c50858d01EXAMPLE dans votre instruction, utilisez un ARN similaire au suivant.

```
"Resource": "arn:aws:ssm:us-east-2:123456789012:maintenancewindow/mw-0c50858d01EXAMPLE"
```

Pour spécifier toutes les fenêtres de maintenance appartenant à un compte spécifique, utilisez le caractère générique (*).

```
"Resource": "arn:aws:ssm:us-east-2:123456789012:maintenancewindow/*"
```

Pour les actions d'API `Parameter Store`, vous pouvez fournir ou restreindre l'accès à tous les paramètres à un niveau d'une hiérarchie à l'aide des noms hiérarchiques et des stratégies AWS Identity and Access Management (IAM), comme suit.

```
"Resource": "arn:aws:ssm:us-west-2:123456789012:parameter/Dev/ERP/Oracle/*"
```

Certaines actions Systems Manager, telles que la création de ressources, ne peuvent pas être exécutées sur une ressource précise. Dans ce cas, vous devez utiliser le caractère générique (*).

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

Certaines actions d'API Systems Manager acceptent plusieurs ressources. Pour spécifier plusieurs ressources dans une seule déclaration, séparez-les ARNs par des virgules comme suit.

```
"Resource": [
      "resource1",
      "resource2"
```

> **Note**
>
> La plupart des AWS services interprètent de la même manière les signes deux points (:) et barre oblique (/).ARNs Cependant, Systems Manager nécessite une correspondance exacte dans les règles et les modèles de ressources. Lors de la création de modèles d'événements, veillez à utiliser les caractères ARN corrects afin qu'ils correspondent à l'ARN de la ressource.

Le tableau suivant décrit les formats ARN des types de ressources pris en charge par Systems Manager.

Type de ressource	Format ARN
Association	arn:aws:ssm :*region*:*account-id*:association/*association-id*
Exécution d'Automation	arn:aws:ssm :*region*:*account-id*:exécution-automatisation/*automation-execution-id*
Définition d'Automation (avec sous-ressource de version)	arn:aws:ssm :*region*:*account-id*:définition-automatisation/*automation-definition-id*:*version-id* (1)
Document	arn:aws:ssm :*region*:*account-id*:document/*document-name*
Fenêtre de maintenance	arn:aws:ssm :*region*:*account-id*:fenêtremaintenance/*window-id*
Instance gérée	arn:aws:ssm :*region*:*account-id*:instance-gérée/*managed-instance-id*
Inventaire des instances gérées	arn:aws:ssm :*region*:*account-id*:inventaire-instance-géré/*managed-instance-id*
Paramètre	Un paramètre de premier niveau : • arn:aws:ssm :*region*:*account-id*:paramètre/*parameter-name*/ Un paramètre nommé avec une construction hiérarchique : • arn:aws:ssm :*region*:*account-id*:paramètre/*parameter-name-root*/*level-2*/*level-3*/*level-4*/*level-5* (2)
Référence de correctif	arn:aws:ssm :*region*:*account-id*:référence de correctif/*patch-baseline-id*
Session	arn:aws:ssm :*region*:*account-id*:session/*session-id* (3)
Toutes les ressources Systems Manager	arn:aws:ssm:*
Toutes les ressources Systems Manager appartenant au compte spécifié dans la région indiquée	arn:aws:ssm :*region*:*account-id*:*

(1) Pour les définitions d'automatisation, Systems Manager prend en charge une ressource de deuxième niveau, l'ID de version. Dans AWS, ces ressources de deuxième niveau sont appelées des sous-ressources. La spécification d'une sous-ressource de version pour une ressource de définition de l'automatisation vous permet de fournir l'accès à certaines versions d'une définition de l'automatisation. Par exemple, il se peut que vous souhaitiez veiller à ce que seule la dernière version de la définition de l'automatisation soit utilisée dans votre gestion des instances.

(2) Pour organiser et gérer des paramètres, vous pouvez créer des noms pour les paramètres à l'aide d'une construction hiérarchique. Grâce à une construction hiérarchique, un nom de paramètre peut inclure un chemin que vous définissez en utilisant des barres obliques. Vous pouvez nommer une ressource de paramètre avec quinze niveaux maximum. Nous vous suggérons de créer des hiérarchies qui reflètent une structure hiérarchique existante dans votre environnement. Pour plus d'informations, consultez la section Création de paramètres Systems Manager (p. 243).

3 Dans la plupart des cas, l'ID de session est construit avec l'ID de l'utilisateur de compte qui a démarré la session, auquel est ajouté un suffixe alphanumérique. Exemple :

```
arn:aws:us-east-2:111122223333:session/JohnDoe-1a2b3c4sEXAMPLE
```

Toutefois, si l'ID utilisateur n'est pas disponible, l'ARN est construit plutôt de la façon suivante :

```
arn:aws:us-east-2:111122223333:session/session-1a2b3c4sEXAMPLE
```

Pour plus d'informations sur le format de ARNs , consultez Amazon Resource Name (ARN) et Espaces de noms Service dans le AWS document .Référence générale d'Amazon Web Services

Pour obtenir la liste des types de Systems Manager ressources et de leurs ARNs , consultez Ressources définies par AWS Systems Manager dans le IAM Guide de l'utilisateur . Pour en savoir plus sur les actions avec laquelle vous pouvez spécifier l'ARN de chaque ressource, consultez Actions définies par AWS Systems Manager.

Clés de condition

L'élément `Condition` (ou le bloc `Condition`) vous permet de spécifier des conditions lorsqu'une instruction est appliquée. L'élément `Condition` est facultatif. Vous pouvez créer des expressions conditionnelles qui utilisent des opérateurs de condition, comme égal ou inférieur, pour faire correspondre la condition de la stratégie aux valeurs de la demande.

Si vous spécifiez plusieurs éléments `Condition` dans une instruction, ou plusieurs clés dans un seul élément `Condition`, AWS les évalue à l'aide d'une opération `AND` logique. Si vous spécifiez plusieurs valeurs pour une seule clé de condition, AWS évalue la condition à l'aide d'une opération `OR` logique. Toutes les conditions doivent être remplies avant que les autorisations associées à l'instruction ne soient accordées.

Vous pouvez aussi utiliser des variables d'espace réservé quand vous spécifiez des conditions. Par exemple, vous pouvez accorder à un utilisateur IAM l'autorisation d'accéder à une ressource uniquement si elle est balisée avec son nom d'utilisateur IAM . Pour de plus amples informations, veuillez consulter Éléments des stratégies IAM : variables et balises dans le Guide de l'utilisateur IAM.

Systems Manager définit son propre ensemble de clés de condition et prend également en charge l'utilisation des clés de condition générales. Pour voir toutes les clés de condition globales AWS, consultez Clés de contexte de condition globales AWS dans le IAM Guide de l'utilisateur.

Pour voir une liste de clés de conditions Systems Manager, consultez Clés de condition pour AWS Systems Manager dans le IAM Guide de l'utilisateur. Pour savoir avec quelles actions et ressources vous pouvez utiliser une clé de condition, consultez Actions définies par AWS Systems Manager.

Pour obtenir des informations sur l'utilisation de la clé de condition `ssm:resourceTag/*`, veuillez consulter les rubriques suivantes :

- Restriction de l'accès aux commandes de niveau racine viaAgent SSM (p. 108)
- Restriction de l'accès de la fonctionnalité Fonctionnalité Exécuter la commande en fonction des balises d'instance (p. 943)
- Limiter l'accès à la session en fonction des balises d'instance (p. 892)

Pour plus d'informations sur l'utilisation des clés de condition `ssm:Recursive` et `ssm:Overwrite`, consultez Utilisation des hiérarchies de paramètres (p. 285).

Examples

Pour voir des exemples de stratégies Systems Manager basées sur l'identité, veuillez consulter Exemples de stratégies basées sur l'identité AWS Systems Manager (p. 1304).

Stratégies Systems Manager basées sur les ressources

D'autres services AWS, tels que Amazon Simple Storage Service, prennent en charge les stratégies d'autorisation basées sur les ressources. Par exemple, vous pouvez attacher une stratégie d'autorisation à un compartiment S3 pour gérer les autorisations d'accès à ce compartiment.

Systems Manager ne prend pas en charge les stratégies basées sur les ressource.

Autorisation basée sur les balises Systems Manager

Vous pouvez attacher des balises aux ressources Systems Manager ou transmettre des balises dans une requête à Systems Manager. Pour contrôler l'accès basé sur des balises, vous devez fournir les informations de balise dans l'élément de condition d'une stratégie utilisant les clés de condition `ssm:resourceTag/key-name`, `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` ou `aws:TagKeys`. Vous pouvez ajouter des balises aux types de ressources suivants lorsque vous les créez ou les mettez à jour :

- Document
- Instance gérée
- Fenêtre de maintenance
- Paramètre
- Référence de correctif
- OpsItem

Pour de plus amples informations sur le balisage des ressources Systems Manager, veuillez consulter Balisage des ressources Systems Manager (p. 1359).

Pour voir un exemple de stratégie basée sur l'identité permettant de limiter l'accès à une ressource en fonction des balises de celle-ci, consultez Affichage de documents Systems Manager basés sur des balises (p. 1309).

Rôles IAM Systems Manager

Un rôle IAM est une entité au sein de votre compte AWS qui dispose d'autorisations spécifiques.

Utilisation d'informations d'identification temporaires avec Systems Manager

Vous pouvez utiliser des informations d'identification temporaires pour vous connecter avec la fédération, assumer un rôle IAM, ou encore pour assumer un rôle entre comptes. Vous obtenez des informations d'identification de sécurité temporaires en appelant des opérations AWS STS d'API comme AssumeRole ou GetFederationToken .

Systems Manager prend en charge l'utilisation des informations d'identification temporaires.

Rôles liés à un service

Les rôles liés à un service permettent aux services AWS d'accéder à des ressources dans d'autres services pour effectuer une action en votre nom. Les rôles liés à un service s'affichent dans votre compte IAM et sont détenus par le service. Un administrateur IAM peut consulter, mais ne peut pas modifier les autorisations concernant les rôles liés à un service.

Systems Manager prend en charge les rôles liés à un service. Pour de plus amples informations sur la création ou la gestion des rôles liés à un service Systems Manager, veuillez consulter Utilisation des rôles liés à un service pour Systems Manager (p. 1312).

Rôles de service

Cette fonction permet à un service d'endosser un rôle de service en votre nom. Ce rôle autorise le service à accéder à des ressources d'autres services pour effectuer une action en votre nom. Les rôles de service s'affichent dans votre compte IAM et sont la propriété du compte. Cela signifie qu'un administrateur IAM peut modifier les autorisations associées à ce rôle. Toutefois, une telle action peut perturber le bon fonctionnement du service.

Systems Manager prend en charge les rôles de service.

Choix d'un rôle IAM dans Systems Manager

Pour que Systems Manager interagisse avec vos instances gérées, vous devez choisir un rôle autorisant Systems Manager à accéder aux instances en votre nom. Si vous avez déjà créé un rôle de service ou un rôle lié à un service, Systems Manager vous fournit une liste de rôles dans laquelle effectuer votre choix. Il est important de choisir un rôle qui permet d'accéder au démarrage et à l'arrêt des instances gérées.

Pour accéder aux instances EC2, le rôle dont votre compte AWS a besoin est un profil d'instance IAM. Pour plus d'informations, consultez la section Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34).

Pour accéder aux instances sur site ou aux machines virtuelles (VM), votre compte AWS a besoin d'un rôle de service IAM pour un environnement hybride. Pour plus d'informations, consultez la section Créer un rôle de service IAM pour un environnement hybride (p. 50).

Un flux de travail Automation peut être lancé dans le contexte d'un rôle de service (ou rôle de responsable). Cela permet au service d'effectuer des actions en votre nom. Si vous ne spécifiez pas de rôle de responsable, Automation utilise le contexte de l'utilisateur qui a appelé l'exécution. Cependant, certaines situations exigent que vous spécifiiez un rôle de service pour Automation. Pour plus d'informations, consultez la section Configuration d'un accès à un rôle de service (rôle de responsable) pour les flux de travail Automation (p. 308).

Stratégies gérées par AWS pour AWS Systems Manager

AWS est adapté à de nombreux cas d'utilisation courants et fournit des stratégies IAM autonomes qui sont créées et administrées par AWS. Ces stratégies gérées AWS accordent les autorisations nécessaires pour les cas d'utilisation courants, afin de vous éviter d'avoir à réfléchir aux autorisations requises. (Vous pouvez également créer vos propres IAM stratégies personnalisées afin d'accorder des autorisations pour les Systems Manager actions et les ressources .) Pour plus d'informations, consultez Stratégies gérées par AWS dans le IAM Guide de l'utilisateur .

Les stratégies gérées par AWS suivantes, que vous pouvez attacher aux utilisateurs de votre compte, sont propres à AWS Systems Manager :

- Stratégie d'approbation d'AmazonSSMFullAccess–utilisateur qui accorde un accès complet à lSystems Manager'API et aux documents.
- Stratégie d'approbation d'AmazonSSMReadOnlyAccess–utilisateur qui accorde l'accès aux actions d'API Systems Manager en lecture seule, telles que `Get*` et `List*` .
- AmazonSSMAutomationApproverAccess–Une stratégie d'approbation d'utilisateur qui permet d'accéder à l'affichage des exécutions d'automatisation et d'envoyer des décisions d'approbation à l'automatisation en attente d'approbation.
- AmazonSSMAutomationRole–La stratégie de rôle de service qui fournit des autorisations pour que le service AWS Systems Manager d'automatisation exécute les activités définies dans les documents d'automatisation. Attribuez cette stratégie aux administrateurs et aux utilisateurs avancés de confiance.
- AmazonSSMMaintenanceWindowRole–Stratégie de rôle de service pour Systems Manager Fenêtres de maintenance .

- Stratégie d'approbation d'AmazonSSMDirectoryServiceAccess–instance qui autorise Agent SSM à accéder au nom de l'utilisateur pour les demandes de jonction du domaine par l'instance gérée.AWS Directory Service
- Stratégie d'approbation d'AmazonSSMManagedInstanceCore–instance qui permet à une instance d'utiliser les fonctionnalités de base du AWS Systems Manager service.
- Stratégie AmazonSSMServiceRolePolicy – de rôle de service qui fournit l'accès aux AWS ressources gérées ou utilisées par AWS Systems Manager .
- Stratégie de rôle de AWSResourceAccessManagerServiceRolePolicy – service contenant un AWS Resource Access Manager accès en lecture seule à la structure AWS Organizations du compte. Il contient également des autorisations IAM pour supprimer le rôle.
- AmazonEC2RoleforSSM–Cette stratégie sera bientôt obsolète. À la place, utilisez la AmazonSSMManagedInstanceCore stratégie pour activer la fonctionnalité de base du AWS Systems Manager service sur les instances EC2. Pour plus d'informations, consultez Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34).

> **Note**
>
> Dans un environnement hybride, vous avez besoin d'un IAM rôle supplémentaire qui autorise les serveurs et VMs à communiquer avec le Systems Manager service . Il s'agit du rôle de service IAM pour Systems Manager. Ce rôle accorde au AWS Security Token Service service AWS STS lAssumeRole'approbation (Systems Manager). L'action `AssumeRole` renvoie un ensemble d'informations d'identification de sécurité temporaires (composé d'un ID de clé d'accès, d'une clé d'accès secrète et d'un jeton de sécurité). Vous utilisez ces informations d'identification temporaires pour accéder aux ressources AWS auxquelles vous n'avez normalement pas accès. Pour de plus amples informations, veuillez consulter Créer un rôle de service IAM pour un environnement hybride (p. 50) et AssumeRole dans AWS Security Token Service API Reference.

Exemples de stratégies basées sur l'identité AWS Systems Manager

Par défaut, les utilisateurs et les rôles IAM ne sont pas autorisés à créer ou à modifier les ressources Systems Manager. Ils ne peuvent pas non plus exécuter des tâches à l'aide de AWS Management Console, AWS CLI ou de l'API AWS. Un administrateur IAM doit créer des stratégies IAM autorisant les utilisateurs et les rôles à exécuter des opérations d'API spécifiques sur les ressources spécifiées dont ils ont besoin. Il doit ensuite attacher ces stratégies aux utilisateurs ou aux groupes IAM ayant besoin de ces autorisations.

L'exemple suivant illustre une stratégie d'autorisations qui permet à un utilisateur de supprimer les documents dont le nom commence par **`MyDocument-`** dans la région **`us-west-2`**.

```
{
  "Version": "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : [
        "ssm:DeleteDocument"
      ],
      "Resource" : [
        "arn:aws:ssm:us-west-2:aws-account-ID:document/MyDocument-*"
      ]
    }
  ]
}
```

Pour apprendre à créer une stratégie basée sur l'identité IAM à l'aide de ces exemples de document de stratégie JSON, consultez Création de stratégies dans l'onglet JSON dans le IAM Guide de l'utilisateur.

Rubriques

- Bonnes pratiques en matière de stratégies (p. 1305)
- Utilisation de la console Systems Manager (p. 1305)
- Autorisation accordée aux utilisateurs pour afficher leurs propres autorisations (p. 1306)
- Exemples de stratégies gérées par le client (p. 1307)
- Affichage de documents Systems Manager basés sur des balises (p. 1309)

Bonnes pratiques en matière de stratégies

Les stratégies basées sur l'identité sont très puissantes. Elles déterminent si une personne peut créer, consulter ou supprimer des ressources Systems Manager dans votre compte. Ces actions peuvent entraîner des frais pour votre compte AWS. Lorsque vous créez ou modifiez des stratégies basées sur l'identité, suivez ces instructions et recommandations :

- Commencer à utiliser des stratégies gérées AWS – Pour commencer à utiliser Systems Manager rapidement, utilisez les politiques gérées AWS pour accorder à vos employés les autorisations dont ils ont besoin. Ces stratégies sont déjà disponibles dans votre compte et sont gérées et mises à jour par AWS. Pour plus d'informations, consultez la section Mise en route avec les autorisations à l'aide des stratégies gérées AWS dans le IAM Guide de l'utilisateur.
- Accorder le privilège le plus faible – Lorsque vous créez des stratégies personnalisées, accordez uniquement les autorisations requises pour exécuter une seule tâche. Commencez avec un minimum d'autorisations et accordez-en d'autres si nécessaire. Cette méthode est plus sûre que de commencer avec des autorisations trop permissives et d'essayer de les restreindre plus tard. Pour plus d'informations, consultez Accorder le privilège le plus faible dans le IAM Guide de l'utilisateur.
- Activer MFA pour les opérations sensibles – Pour plus de sécurité, obligez les utilisateurs IAM à utiliser l'authentification multi-facteurs (MFA) pour accéder à des ressources ou à des opérations d'API sensibles. Pour plus d'informations, consultez Utilisation de Multi-Factor Authentication (MFA) dans AWS dans le IAM Guide de l'utilisateur.
- Utiliser des conditions de stratégie pour une plus grande sécurité – Tant que cela reste pratique pour vous, définissez les conditions dans lesquelles vos stratégies basées sur l'identité autorisent l'accès à une ressource. Par exemple, vous pouvez rédiger les conditions pour spécifier une plage d'adresses IP autorisées d'où peut provenir une demande. Vous pouvez également écrire des conditions pour autoriser les requêtes uniquement à une date ou dans une plage de temps spécifiée, ou pour imposer l'utilisation de SSL ou de MFA. Pour plus d'informations, consultez Éléments de stratégie JSON IAM : condition dans le IAM Guide de l'utilisateur.

Utilisation de la console Systems Manager

Pour accéder à la console AWS Systems Manager, vous devez détenir un ensemble minimum d'autorisations. Ces autorisations doivent vous permettre de répertorier et consulter les informations relatives aux ressources Systems Manageret aux autres ressources de votre compte AWS.

Pour exploiter pleinement Systems Manager dans la console Systems Manager, vous devez disposer des autorisations provenant des services suivants :

- AWS Systems Manager
- Amazon Elastic Compute Cloud (Amazon EC2)
- AWS Identity and Access Management (IAM)

Vous pouvez accorder les autorisations requises avec la déclaration de stratégie suivante.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:*",
                "ec2:describeInstances",
                "iam:PassRole",
                "iam:ListRoles"
            ],
            "Resource": "*"
        }
    ]
}
```

Si vous créez une stratégie 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 et rôles IAM) tributaires de cette stratégie.

Vous n'avez pas besoin d'accorder les autorisations minimales de console aux utilisateurs qui effectuent des appels uniquement vers l'AWS CLI ou l'API AWS. Autorisez plutôt l'accès à uniquement aux actions qui correspondent à l'opération d'API que vous tentez d'effectuer.

Autorisation accordée aux utilisateurs pour afficher leurs propres autorisations

Cet exemple montre comment créer une stratégie qui permet aux utilisateurs IAM d'afficher les stratégies en ligne et gérées attachées à leur identité d'utilisateur. Cette stratégie inclut les autorisations nécessaires pour réaliser cette action sur la console ou par programmation à l'aide de l'AWS CLI ou de l'API 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": "*"
        }
```

```
    ]
}
```

Exemples de stratégies gérées par le client

Vous pouvez créer des stratégies autonomes que vous gérez dans votre propre compte AWS. Nous les appelons stratégies gérées par le client. Vous pouvez attacher ces stratégies à plusieurs entités principales dans votre compte AWS. Lorsque vous attachez une stratégie à une entité mandataire, vous accordez à cette dernière les autorisations définies dans la stratégie. Pour de plus amples informations, veuillez consulter Stratégies gérées par le client dans le document IAM Guide de l'utilisateur.

Les exemples suivants de stratégies utilisateur accordent des autorisations pour différentes actions AWS Systems Manager. Utilisez-les pour limiter l'accès à Systems Manager pour vos utilisateurs et rôles IAM. Ces stratégies fonctionnent lors de l'exécution dSystems Manager'actions dans l'API AWS SDKs , l' ou l' AWS CLI . Pour les utilisateurs qui utilisent la console, vous devez accorder des autorisations supplémentaires propres à la console. Pour plus d'informations, consultez la section Utilisation de la console Systems Manager (p. 1305).

> **Note**
>
> Tous les exemples utilisent la Région USA Ouest (Oregon) région (us-west-2) et contiennent un compte fictifIDs.

Exemples

- Exemple 1 : Permettre à un utilisateur d'effectuer des Systems Manager opérations dans une seule région (p. 1307)
- Exemple 2 : Autoriser un utilisateur à répertorier les documents pour une seule région (p. 1307)

Exemple 1 : Permettre à un utilisateur d'effectuer des Systems Manager opérations dans une seule région

L'exemple suivant accorde des autorisations pour effectuer des opérations AWS Systems Manager uniquement dans la région **us-west-2** :

```
{
  "Version": "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : [
        "ssm:*"
      ],
      "Resource" : [
        "arn:aws:ssm:us-west-2:aws-account-ID:*"
      ]
    }
  ]
}
```

Exemple 2 : Autoriser un utilisateur à répertorier les documents pour une seule région

Dans l'exemple suivant, des autorisations sont accordées pour répertorier tous les noms de document qui commencent par **Update** dans la région **us-west-2** :

```
{
  "Version": "2012-10-17",
```

```
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : [
        "ssm:ListDocuments"
      ],
      "Resource" : [
        "arn:aws:ssm:us-west-2:aws-account-ID:document/Update*"
      ]
    }
  ]
}
```

Exemple 3 : Autoriser un utilisateur à utiliser un document SSM spécifique pour exécuter des commandes sur des instances spécifiques

L'exemple de stratégie IAM qui suit autorise un utilisateur à effectuer les opérations suivantes.

- Lister des documents et des versions de documents Systems Manager.
- Afficher les détails des documents.
- Envoyer une commande à l'aide du document indiqué dans la stratégie. Le nom du document est défini par cette entrée :

```
arn:aws:ssm:us-east-2:aws-account-ID:document/Systems-Manager-document-name
```

- Envoyer une commande à trois instances. Les instances sont déterminées par les entrées suivantes dans la seconde section `Resource` :

```
"arn:aws:ec2:us-east-2:aws-account-ID:instance/i-02573cafcfEXAMPLE",
"arn:aws:ec2:us-east-2:aws-account-ID:instance/i-0471e04240EXAMPLE",
"arn:aws:ec2:us-east-2:aws-account-ID:instance/i-07782c72faEXAMPLE"
```

- Afficher les détails d'une commande après qu'elle ait été envoyée.
- Démarrer et arrêter les exécutions d'automatisation.
- Obtenir des informations sur les exécutions d'automatisation.

Si vous souhaitez accorder à un utilisateur l'autorisation d'utiliser ce document pour envoyer des commandes sur n'importe quelle instance à laquelle l'utilisateur a actuellement accès (tel que déterminé par son compte utilisateur AWS), vous pouvez spécifier l'entrée suivante dans la section `Resource` et supprimer les autres entrées d'instance.

```
"arn:aws:ec2:us-east-2:*:instance/*"
```

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "ssm:ListDocuments",
                "ssm:ListDocumentVersions",
                "ssm:DescribeDocument",
                "ssm:GetDocument",
                "ssm:DescribeInstanceInformation",
                "ssm:DescribeDocumentParameters",
                "ssm:DescribeInstanceProperties"
            ],
            "Effect": "Allow",
            "Resource": "*"
```

```
        },
        {
            "Action": "ssm:SendCommand",
            "Effect": "Allow",
            "Resource": [
                "arn:aws:ec2:us-east-2:aws-account-ID:instance/i-02573cafcfEXAMPLE",
                "arn:aws:ec2:us-east-2:aws-account-ID:instance/i-0471e04240EXAMPLE",
                "arn:aws:ec2:us-east-2:aws-account-ID:instance/i-07782c72faEXAMPLE",

                "arn:aws:ssm:us-east-2:aws-account-ID:document/Systems-Manager-document-
name"
            ]
        },
        {
            "Action": [
                "ssm:CancelCommand",
                "ssm:ListCommands",
                "ssm:ListCommandInvocations"
            ],
            "Effect": "Allow",
            "Resource": "*"
        },
        {
            "Action": "ec2:DescribeInstanceStatus",
            "Effect": "Allow",
            "Resource": "*"
        },
        {
            "Action": "ssm:StartAutomationExecution",
            "Effect": "Allow",
            "Resource": [
                "arn:aws:ssm:::automation-definition/"
            ]
        },
        {
            "Action": "ssm:DescribeAutomationExecutions ",
            "Effect": "Allow",
            "Resource": [
                "*"
            ]
        },
        {
            "Action": [
                "ssm:StopAutomationExecution",
                "ssm:GetAutomationExecution"
            ],
            "Effect": "Allow",
            "Resource": [
                "arn:aws:ssm:::automation-execution/"
            ]
        }
    ]
}
```

Affichage de documents Systems Manager basés sur des balises

Vous pouvez utiliser des conditions dans votre stratégie basée sur l'identité pour contrôler l'accès aux ressources Systems Manager en fonction des balises. Cet exemple montre comment créer une stratégie qui permet d'afficher un document SSM. Toutefois, l'autorisation est accordée uniquement si la balise de document `Owner` a la valeur du nom de l'utilisateur. Cette stratégie accorde également les autorisations nécessaires pour réaliser cette action sur la console.

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ListDocumentsInConsole",
            "Effect": "Allow",
            "Action": "ssm:ListDocuments",
            "Resource": "*"
        },
        {
            "Sid": "ViewDocumentIfOwner",
            "Effect": "Allow",
            "Action": "ssm:GetDocument",
            "Resource": "arn:aws:ssm:*:*:document/*",
            "Condition": {
                "StringEquals": {"ssm:ResourceTag/Owner": "${aws:username}"}
            }
        }
    ]
}
```

Vous pouvez attacher cette stratégie aux utilisateurs IAM de votre compte. Si un utilisateur nommé `richard-roe` tente d'afficher un Systems Manager document , celui-ci doit être balisé `Owner=richard-roe` ou `owner=richard-roe` . Dans le cas contraire, l'utilisateur se voit refuser l'accès. La clé de condition de balise `Owner` correspond à la fois à `Owner` et à `owner`, car les noms de clé de condition ne sont pas sensibles à la casse. Pour plus d'informations, consultez Éléments de stratégie IAM JSON : Condition dans le IAM Guide de l'utilisateur .

Résolution des problèmes d'identité et d'accès AWS Systems Manager

Consultez les informations suivantes pour identifier et résoudre les problèmes courants que vous pouvez rencontrer lorsque vous travaillez avec Systems Manager et IAM.

Rubriques

- Je ne suis pas autorisé à effectuer une action dans Systems Manager (p. 1310)
- Je ne suis pas autorisé à exécuter iam:PassRole (p. 1311)
- Je veux afficher mes clés d'accès (p. 1311)
- Je suis un administrateur et je souhaite autoriser d'autres utilisateurs à accéder à Systems Manager (p. 1311)
- Je souhaite permettre à des personnes extérieures à mon compte AWS d'accéder à mes ressources Systems Manager (p. 1312)

Je ne suis pas autorisé à effectuer une action dans Systems Manager

Si AWS Management Console indique que vous n'êtes pas autorisé à exécuter une action, vous devez contacter votre administrateur pour obtenir de l'aide. Votre administrateur est la personne qui vous a fourni votre nom d'utilisateur et votre mot de passe.

L'exemple d'erreur suivant se produit lorsque l'utilisateur IAM `mateojackson` tente d'utiliser la console pour afficher des informations détaillées sur un document, mais qu'il ne dispose pas des autorisations `ssm:GetDocument` nécessaires.

```
User: arn:aws:ssm::123456789012:user/mateojackson is not authorized to perform:
 ssm:GetDocument on resource: MyExampleDocument
```

Dans ce cas, Mateo demande à son administrateur de mettre à jour ses stratégies pour l'autoriser à accéder à la ressource `MyExampleDocument` à l'aide de l'action `ssm:GetDocument`.

Je ne suis pas autorisé à exécuter iam:PassRole

Si vous recevez un message d'erreur selon lequel vous n'êtes pas autorisé à exécuter l'action `iam:PassRole`, vous devez contacter votre administrateur pour obtenir de l'aide. Votre administrateur est la personne qui vous a fourni votre nom d'utilisateur et votre mot de passe. Demandez à cette personne de mettre à jour vos stratégies pour vous permettre de transmettre un rôle à Systems Manager.

Certains services AWS vous permettent de transmettre un rôle existant à ce service, au lieu de créer un nouveau rôle de service ou rôle lié à un service. Pour ce faire, un utilisateur doit 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 exécuter une action dans Systems Manager. Toutefois, l'action nécessite que le service ait des autorisations accordées par un rôle de service. Mary ne dispose pas des autorisations nécessaires pour transférer le rôle au service.

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

Dans ce cas, Mary demande à son administrateur de mettre à jour ses stratégies pour lui permettre d'exécuter l'action `iam:PassRole`.

Je veux afficher mes clés d'accès

Une fois que vous avez créé vos clés d'accès utilisateur IAM, vous pouvez afficher votre ID de clé d'accès à tout moment. Toutefois, vous ne pouvez pas afficher à nouveau votre clé d'accès secrète. Si vous perdez votre clé d'accès secrète, vous devez créer une nouvelle paire de clés.

Les clés d'accès se composent de deux parties : un ID de clé d'accès (par exemple, `AKIAIOSFODNN7EXAMPLE`) et une clé d'accès secrète (par exemple, `wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY`). À l'instar d'un nom d'utilisateur et un mot de passe, vous devez utiliser à la fois l'ID de clé d'accès et la clé d'accès secrète pour authentifier vos demandes. Gérez vos clés d'accès de manière aussi sécurisée que votre nom d'utilisateur et votre mot de passe.

> **Important**
>
> Ne communiquez pas vos clés d'accès à un tiers, même pour qu'il vous aide à trouver votre ID utilisateur canonique. En effet, vous lui accorderiez ainsi un accès permanent à votre compte.

Lorsque vous créez une paire de clé d'accès, enregistrez l'ID de clé d'accès et la clé d'accès secrète dans un emplacement sécurisé. La clé d'accès secrète est accessible uniquement au moment de sa création. Si vous perdez votre clé d'accès secrète, vous devez ajouter de nouvelles clés d'accès pour votre utilisateur IAM. Vous pouvez avoir un maximum de deux clés d'accès. Si vous en avez déjà deux, vous devez supprimer une paire de clés avant d'en créer une nouvelle. Pour afficher les instructions, consultez Gestion des clés d'accès dans le IAM Guide de l'utilisateur.

Je suis un administrateur et je souhaite autoriser d'autres utilisateurs à accéder à Systems Manager

Pour permettre à d'autres utilisateurs d'accéder à Systems Manager, vous devez créer une entité IAM (utilisateur ou rôle) pour la personne ou l'application qui a besoin de l'accès. Ils utiliseront les informations d'identification de cette entité pour accéder à AWS. Vous devez ensuite associer une stratégie à l'entité qui leur accorde les autorisations appropriées dans Systems Manager.

Pour démarrer immédiatement, consultez Création de votre premier groupe et utilisateur délégué IAM dans le IAM Guide de l'utilisateur.

Je souhaite permettre à des personnes extérieures à mon compte AWS d'accéder à mes ressources Systems Manager

Vous pouvez créer un rôle que les utilisateurs provenant d'autres comptes ou les personnes extérieures à votre organisation peuvent utiliser pour accéder à vos ressources. Vous pouvez spécifier qui est approuvé pour assumer le rôle. Pour les services qui prennent en charge les stratégies basées sur les ressources ou les listes de contrôle d'accès (ACL), vous pouvez utiliser ces stratégies pour accorder aux personnes l'accès à vos ressources.

Pour en savoir plus, consultez les éléments suivants :

- Pour savoir si Systems Manager prend en charge ces fonctionnalités, consultez Fonctionnement d'AWS Systems Manager avec IAM (p. 1298).
- Pour savoir comment fournir un accès à vos ressources sur les comptes AWS que vous détenez, consultez Octroi à un utilisateur IAM de l'autorisation d'accès à un autre compte AWS vous appartenant dans le IAM Guide de l'utilisateur.
- Pour savoir comment fournir l'accès à vos ressources à des comptes AWS tiers, consultez Octroi d'un accès à des comptes AWS appartenant à des tiers dans le IAM Guide de l'utilisateur.
- Pour savoir comment fournir un accès par le biais de la fédération d'identité, consultez Octroi d'accès à des utilisateurs authentifiés en externe (fédération d'identité) dans le IAM Guide de l'utilisateur.
- Pour en savoir plus sur la différence entre l'utilisation des rôles et des stratégies basées sur les ressources pour l'accès entre comptes, consultez Différence entre les rôles IAM et les stratégies basées sur les ressources dans le IAM Guide de l'utilisateur.

Utilisation des rôles liés à un service pour Systems Manager

AWS Systems Manager utilise les rôles liés à un service AWS Identity and Access Management (IAM). Un rôle lié à un service est un type unique de rôle IAM lié directement à Systems Manager. Les rôles liés à un service sont prédéfinis par Systems Manager et comprennent toutes les autorisations nécessaires au service pour appeler d'autres services AWS en votre nom.

Rubriques

- Utilisation des rôles pour collecter l'inventaire et exécuter les tâches de la fenêtre de maintenance (p. 1312)
- Utilisation des rôles pour collecter des informations de compte AWS pour Systems Manager Explorer (p. 1315)

Utilisation des rôles pour collecter l'inventaire et exécuter les tâches de la fenêtre de maintenance

AWS Systems Manager utilise les rôles liés à un service AWS Identity and Access Management (IAM). Un rôle lié à un service est un type unique de rôle IAM lié directement à Systems Manager. Les rôles liés à un service sont prédéfinis par Systems Manager et comprennent toutes les autorisations nécessaires au service pour appeler d'autres services AWS en votre nom.

Un rôle lié à un service permet d'utiliser Systems Manager plus facilement, car vous n'avez pas besoin d'ajouter manuellement les autorisations requises. Systems Manager définit les autorisations de ses rôles liés à un service et, sauf définition contraire, seul Systems Manager peut endosser ses rôles. Les

autorisations définies comprennent la stratégie d'approbation et la stratégie d'autorisation. De plus, cette stratégie d'autorisation ne peut pas être attachée à une autre entité IAM.

Pour de plus amples informations sur les autres services qui prennent en charge les rôles liés à un service, veuillez consulter Services AWS qui fonctionnent avec IAM et recherchez les services qui comportent Oui dans la colonne Rôle lié à un service. Choisissez un Oui ayant un lien permettant de consulter la documentation du rôle lié à un service, pour ce service.

Autorisations des rôles liés à un service pour Systems Manager

Systems Manager utilise le rôle lié à un service nommé AWSServiceRoleForAmazonSSM – AWS Systems Manager utilise ce rôle de service IAM pour gérer les ressources AWS en votre nom..

Le rôle lié à un service AWSServiceRoleForAmazonSSM fait confiance uniquement à ssm.amazonaws.com pour assumer le rôle.

Actuellement, seules deux fonctionnalités Systems Manager utilisent le rôle lié à un service :

- L'inventaire nécessite un rôle lié à un service. Le rôle permet au système de collecter les métadonnées d'inventaires à partir des balises et des groupes de ressources.
- La fonctionnalité Fenêtres de maintenance peut éventuellement utiliser le rôle lié à un service. Ce rôle permet au service Fenêtres de maintenance d'exécuter des tâches de maintenance sur les instances cibles. Notez que le rôle lié au service Systems Manager ne fournit pas les autorisations nécessaires pour tous les scénarios. Pour plus d'informations, veuillez consulter Dois-je utiliser un rôle lié à un service ou un rôle de service personnalisé pour exécuter des tâches de fenêtre de maintenance ? (p. 691)

Sauf mention contraire, la stratégie des autorisations d'un rôle lié à un service AWSServiceRoleForAmazonSSM permet à Systems Manager d'effectuer les actions suivantes sur toutes les ressources connexes (`"Resource": "*"`) :

- `ssm:CancelCommand`
- `ssm:GetCommandInvocation`
- `ssm:ListCommandInvocations`
- `ssm:ListCommands`
- `ssm:SendCommand`
- `ssm:GetAutomationExecution`
- `ssm:GetParameters`
- `ssm:StartAutomationExecution`
- `ssm:ListTagsForResource`
- `ec2:DescribeInstanceAttribute`
- `ec2:DescribeInstanceStatus`
- `ec2:DescribeInstances`
- `lambda:InvokeFunction`[1]
- `states:DescribeExecution`[2]
- `states:StartExecution`[2]
- `resource-groups:ListGroups`
- `resource-groups:ListGroupResources`
- `resource-groups:GetGroupQuery`
- `cloudformation:DescribeStacks`
- `config:SelectResourceConfig`

- `compute-optimizer:GetEC2InstanceRecommendations`
- `support:DescribeTrustedAdvisorChecks`
- `support:DescribeTrustedAdvisorCheckSummaries`
- `support:DescribeTrustedAdvisorCheckResult`
- `iam:PassRole`[3]

[1] L'action `lambda:InvokeFunction` n'est autorisée que pour les ressources suivantes :

```
arn:aws:lambda:*:*:function:SSM*
arn:aws:lambda:*:*:function:*:SSM*
```

[2] Les actions `states:` ne sont autorisées que pour les ressources suivantes :

```
arn:aws:states:*:*:stateMachine:SSM*
arn:aws:states:*:*:execution:SSM*
```

[3] L'action `iam:PassRole` n'est autorisée par la condition suivante que pour le service Systems Manager :

```
"Condition": {
   "StringEquals": {
      "iam:PassedToService": [
         "ssm.amazonaws.com"
      ]
   }
}
```

Création d'un rôle lié à un service pour Systems Manager

Vous pouvez utiliser la console IAM pour créer un rôle lié au service avec le cas d'utilisation EC2. Dans l'interface de ligne de commande IAM ou l'API IAM, créez un rôle lié au service avec le nom de service `ssm.amazonaws.com`. Pour plus d'informations, consultez Création d'un rôle lié à un service dans le IAM Guide de l'utilisateur.

Pour les fenêtres de maintenance uniquement, vous n'avez pas besoin de créer manuellement un rôle lié à un service. Lorsque vous créez une tâche de fenêtre de maintenance dans AWS Management Console, AWS CLI ou l'API Systems Manager, Systems Manager crée pour vous le rôle lié à un service si vous choisissez de ne pas fournir de rôle de service personnalisé.

Si vous supprimez ce rôle lié à un service et que vous avez ensuite besoin de le recréer, vous pouvez utiliser la même procédure pour recréer le rôle dans votre compte.

Modification d'un rôle lié à un service pour Systems Manager

Systems Manager ne vous permet pas de modifier le rôle lié à un service AWSServiceRoleForAmazonSSM. Une fois que vous avez créé un rôle lié à un service, vous ne pouvez pas modifier le nom du rôle, car plusieurs entités peuvent faire référence à ce rôle. Néanmoins, vous pouvez modifier la description du rôle à l'aide de IAM. Pour plus d'informations, consultez la section Modification d'un rôle lié à un service du IAM Guide de l'utilisateur.

Suppression d'un rôle lié à un service pour Systems Manager

Si vous n'avez plus besoin d'utiliser de fonction ni de service nécessitant 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. Vous pouvez utiliser la console IAM, l'interface de ligne de commande

IAM ou l'API IAM pour supprimer manuellement le rôle lié à un service. Pour cela, vous devez commencer par nettoyer manuellement les ressources pour votre rôle lié à un service. Vous pouvez ensuite supprimer manuellement ce rôle.

Comme le rôle lié à un service Systems Manager peut être utilisé par les deux fonctionnalités d'inventaire et de Fenêtres de maintenance, avant d'essayer de supprimer le rôle, assurez-vous qu'aucune d'elles ne l'utilise.

- Inventaire : Si vous supprimez le rôle lié à un service utilisé par l'inventaire Systems Manager, les données d'inventaire pour les balises et les groupes de ressources ne seront plus synchronisés. Vous devez nettoyer les ressources de votre rôle lié à un service avant de pouvoir les supprimer manuellement.
- Fenêtres de maintenance : Vous ne pouvez pas supprimer le rôle lié à un service si des tâches de fenêtre de maintenance s'appuient actuellement sur celui-ci. Vous devez commencer par supprimer le rôle lié à un service des tâches avant de pouvoir supprimer le rôle.

> **Note**
>
> Si le service Systems Manager utilise le rôle lorsque vous essayez de supprimer les balises, des groupes de ressources ou des tâches de fenêtre de maintenance, la suppression peut échouer. Si cela se produit, patientez quelques minutes et réessayez.

Pour supprimer des ressources Systems Manager utilisées par AWSServiceRoleForAmazonSSM.

1. Pour supprimer des balises, consultez Ajout et suppression de balises sur une ressource individuelle.
2. Pour supprimer des groupes de ressources, consultez Suppression des groupes à partir des groupes de ressources AWS.
3. Pour de plus amples informations sur la suppression de tâches de fenêtre de maintenance, veuillez consulter Mettre à jour ou supprimer les tâches de fenêtre de maintenance (console) (p. 715).

Pour supprimer manuellement le rôle lié à un service à l'aide d' IAM

Utilisez la console IAM, l'interface de ligne de commande IAM ou l'API IAM pour supprimer le rôle lié à un service AWSServiceRoleForAmazonSSM. Pour plus d'informations, consultez Suppression d'un rôle lié à un service dans le IAM Guide de l'utilisateur.

Régions prises en charge pour les rôles liés à un service Systems Manager

Systems Manager prend en charge l'utilisation des rôles liés à un service dans toutes les régions où le service est disponible. Pour plus d'informations, consultez la section Régions et points de terminaison AWS pour Systems Manager.

Utilisation des rôles pour collecter des informations de compte AWS pour Systems Manager Explorer

AWS Systems Manager utilise les rôles liés à un service AWS Identity and Access Management (IAM). Un rôle lié à un service est un type unique de rôle IAM lié directement à Systems Manager. Les rôles liés à un service sont prédéfinis par Systems Manager et comprennent toutes les autorisations nécessaires au service pour appeler d'autres services AWS en votre nom.

Un rôle lié à un service permet d'utiliser Systems Manager plus facilement, car vous n'avez pas besoin d'ajouter manuellement les autorisations requises. Systems Manager définit les autorisations de ses

rôles liés à un service et, sauf définition contraire, seul Systems Manager peut endosser ses rôles. Les autorisations définies comprennent la stratégie d'approbation et la stratégie d'autorisation. De plus, cette stratégie 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. Vos ressources Systems Manager sont ainsi protégées, car vous ne pouvez pas involontairement supprimer l'autorisation d'accéder aux ressources.

Pour plus d'informations sur les autres services qui prennent en charge les rôles liés à un service, consultez Services AWS qui fonctionnent avec IAM et recherchez les services qui comportent Oui dans la colonne Rôle lié à un service. Choisissez un Oui ayant un lien permettant de consulter la documentation du rôle lié à un service, pour ce service.

Autorisations des rôles liés à un service pour Systems Manager

Systems Manager utilise le rôle lié à un service nommé AWSServiceRoleForAmazonSSM_AccountDiscovery – AWS Systems Manager utilise ce rôle de service IAM pour appeler d'autres services AWS afin de découvrir les informations de compte AWS..

Le rôle lié à un service AWSServiceRoleForAmazonSSM_AccountDiscovery approuve le fait que les services suivants endossent le rôle :

- `accountdiscovery.ssm.amazonaws.com`

La stratégie d'autorisations liée au rôle permet à Systems Manager d'exécuter les actions suivantes sur les ressources spécifiées :

- organizations:DescribeAccount
- organizations:DescribeOrganization
- organizations:ListAccounts
- organizations:ListAWSServiceAccessForOrganization
- organizations:ListChildren
- organizations:ListParents

Vous devez configurer les autorisations de manière à permettre à une entité IAM (comme un utilisateur, groupe ou rôle) de créer, modifier ou supprimer un rôle lié à un service. Pour de plus amples informations, veuillez consulter Autorisations de rôles liés à un service dans le IAM Guide de l'utilisateur.

Création d'un rôle lié à un service pour Systems Manager

Vous n'avez pas besoin de créer manuellement un rôle lié à un service. Lorsque vous créez une synchronisation des données de ressource à l'aide d' Systems Manager Explorer dans l' AWS Management Console , IAWS CLI' ou l'API AWS , Systems Manager crée automatiquement le rôle lié au service.

Si vous supprimez ce rôle lié à un service et que vous avez ensuite besoin de le recréer, vous pouvez utiliser la même procédure pour recréer le rôle dans votre compte. Lorsque vous créez une synchronisation des données de ressource à l'aide Systems Manager Explorer d' , Systems Manager crée à nouveau le rôle lié au service pour vous.

Modification d'un rôle lié à un service pour Systems Manager

Systems Manager ne vous permet pas de modifier le rôle lié à un service AWSServiceRoleForAmazonSSM_AccountDiscovery. Une fois que vous avez créé un rôle lié à un service, vous ne pouvez pas modifier le nom du rôle, car plusieurs entités peuvent faire référence à ce rôle.

Néanmoins, vous pouvez modifier la description du rôle à l'aide de IAM. Pour plus d'informations, consultez la section Modification d'un rôle lié à un service du IAM Guide de l'utilisateur.

Suppression d'un rôle lié à un service pour Systems Manager

Si vous n'avez plus besoin d'utiliser une fonction 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 votre rôle lié à un service avant de pouvoir le supprimer manuellement.

Nettoyage d'un rôle lié à un service

Pour pouvoir utiliser IAM pour supprimer un rôle lié à un service, vous devez d'abord supprimer toutes les synchronisations de données de ressources Explorer. Pour plus d'informations, consultez la section Suppression d'une synchronisation des données de ressource Systems ManagerExplorer (p. 172).

> Note
>
> Si le service Systems Manager utilise le rôle lorsque vous essayez de supprimer les ressources, la suppression peut échouer. Si cela se produit, patientez quelques minutes et réessayez.

Suppression manuelle du rôle lié à un service

Utilisez la console IAM, l'AWS CLI ou l'API AWS pour supprimer le rôle lié à un service AWSServiceRoleForAmazonSSM_AccountDiscovery. Pour plus d’informations, consultez Suppression d'un rôle lié à un service dans le IAM Guide de l'utilisateur.

Régions prises en charge pour les rôles liés à un service Systems Manager

Systems Manager prend en charge l’utilisation des rôles liés à un service dans toutes les régions où le service est disponible. Pour plus d'informations, consultez la section Régions et points de terminaison AWS.

Systems Manager ne prend pas en charge l'utilisation des rôles liés à un service dans toutes les régions où le service est disponible. Vous pouvez utiliser le rôle AWSServiceRoleForAmazonSSM_AccountDiscovery dans les régions suivantes :

Nom de la région	Identité de la région	Prise en charge dans Systems Manager
USA Est (Virginie du Nord)	us-east-1	Oui
USA Est (Ohio)	us-east-2	Oui
USA Ouest (Californie du Nord)	us-west-1	Oui
USA Ouest (Oregon)	us-west-2	Oui
Asie-Pacifique (Mumbai)	ap-south-1	Oui
Asie-Pacifique (Osaka-Local)	ap-northeast-3	Oui
Asie-Pacifique (Séoul)	ap-northeast-2	Oui
Asie-Pacifique (Singapour)	ap-southeast-1	Oui

Nom de la région	Identité de la région	Prise en charge dans Systems Manager
Asie-Pacifique (Sydney)	ap-southeast-2	Oui
Asie-Pacifique (Tokyo)	ap-northeast-1	Oui
Canada (Centre)	ca-central-1	Oui
Europe (Francfort)	eu-central-1	Oui
Europe (Irlande)	eu-west-1	Oui
Europe (Londres)	eu-west-2	Oui
Europe (Paris)	eu-west-3	Oui
Amérique du Sud (São Paulo)	sa-east-1	Oui
AWS GovCloud (US)	us-gov-west-1	Non

Journalisation et surveillance dans AWS Systems Manager

La surveillance est un enjeu important pour assurer la fiabilité, la disponibilité et les performances d'Systems Manager et de vos solutions AWS. Vous devez collecter les données de surveillance de toutes les parties de votre solution AWS afin de pouvoir déboguer plus facilement une éventuelle défaillance à plusieurs points. AWS fournit plusieurs outils pour surveiller vos ressources Systems Manager et autres, et répondre aux incidents potentiels.

Journaux AWS CloudTrail

CloudTrail fournit un enregistrement des actions réalisées par un utilisateur, un rôle ou un service AWS dans Systems Manager. À l'aide des informations collectées par CloudTrail, vous pouvez déterminer la demande qui a été envoyée à Systems Manager, l'adresse IP source à partir de laquelle la demande a été effectuée, l'auteur de la demande et la date de la demande, ainsi que d'autres informations. Pour plus d'informations, consultez la section Journalisation des appels d'API AWS Systems Manager avec AWS CloudTrail (p. 1336).

Alarmes Amazon CloudWatch

À l'aide des alarmes Amazon CloudWatch, vous surveillez une seule métrique sur une période que vous spécifiez pour vos instances EC2 et autres ressources. Si la métrique dépasse un seuil donné, une notification est envoyée à une rubrique Amazon SNS ou à une stratégie AWS Auto Scaling. Les alarmes CloudWatch n'appellent pas d'actions parce qu'elles sont à un état particulier. L'état doit avoir changé et avoir été conservé pendant un nombre de périodes spécifié. Pour plus d'informations, consultez Utilisation des alarmes Amazon CloudWatch dans le Guide de l'utilisateur Amazon CloudWatch.

Tableaux de bord Amazon CloudWatch

Les tableaux de bord CloudWatch sont des pages d'accueil personnalisables de la console CloudWatch que vous pouvez utiliser pour surveiller vos ressources dans une seule vue, même les ressources réparties dans différentes régions. Vous pouvez utiliser des tableaux de bord CloudWatch pour créer des vues personnalisées des métriques et alarmes de vos ressources AWS. Les tableaux de bord Amazon CloudWatch sont des pages d'accueil personnalisables de la console CloudWatch que vous pouvez utiliser pour surveiller vos ressources dans une seule vue, même les ressources

réparties dans différentes régions. Vous pouvez utiliser les tableaux de bord CloudWatch pour créer des vues personnalisées des métriques et alarmes de vos ressources AWS. Pour plus d'informations, consultez la section Tableaux de bord Amazon CloudWatch hébergés par Systems Manager (p. 210).

Amazon EventBridge

EventBridge vous permet de configurer des règles afin d'être averti des modifications des ressources Systems Manager et pour indiquer à EventBridge d'effectuer des actions en fonction du contenu de ces événements. EventBridge prend en charge plusieurs événements qui sont émis par diverses fonctionnalités Systems Manager. Pour plus d'informations, consultez la section Surveillance des événements Systems Manager avec Amazon EventBridge (p. 1340).

Amazon CloudWatch Logs et journaux de l'Agent SSM

L'agent SSM écrit des informations relatives aux exécutions, actions planifiées, erreurs et états d'intégrité dans des fichiers journaux sur chaque instance. Vous pouvez afficher les fichiers journaux en vous connectant manuellement à une instance. Nous vous recommandons d'envoyer automatiquement les données de journal de l'agent à un groupe de journaux dans CloudWatch Logs à des fins d'analyse. Pour de plus amples informations, veuillez consulter Envoi de journaux d'instance à CloudWatch Logs (Agent CloudWatch) (p. 1327) et Affichage des Agent SSM journaux (p. 107).

Conformité de la configuration AWS Systems Manager

Vous pouvez utiliser le service Conformité de configuration AWS Systems Manager pour analyser votre parc d'instances gérées afin de rechercher des incohérences de conformité et de configuration de correctifs. Vous pouvez collecter et regrouper des données de plusieurs comptes et régions AWS, puis explorer plus en détails des ressources spécifiques qui ne sont pas conformes. Par défaut, le service Conformité de configuration affiche les données de conformité actuelles relatives Systems Manager Gestionnaire de correctifs aux correctifs et Systems Manager State Manager aux associations. Pour plus d'informations, consultez la section Conformité de la configuration AWS Systems Manager (p. 778).

AWS Systems Manager Explorer

Explorer est un tableau de bord des opérations personnalisable qui présente des informations sur vos ressources AWS. Explorer affiche une vue agrégée des données d'opérations (OpsData) pour vos comptes AWS et pour toutes les régions. Dans Explorer , OpsData inclut des métadonnées sur vos instances EC2, des détails de conformité des correctifs et des éléments de travail opérationnels (OpsItems). Explorer fournit un contexte sur la répartition entre vos unités commerciales ou applications, leur évolution au fil du temps et leur variation par catégorie.OpsItems Vous pouvez regrouper et filtrer les informations dans Explorer pour vous concentrer sur les éléments qui vous intéressent et qui nécessitent une action. Pour plus d'informations, consultez la section AWS Systems Manager Explorer (p. 153).

AWS Systems Manager OpsCenter

OpsCenterfournit un emplacement central où les ingénieurs d'exploitation et les professionnels de l'informatique peuvent afficher, étudier et résoudre des éléments de travail opérationnels (OpsItems) liés aux ressources AWS. OpsCenter regroupe et standardise OpsItems les différents services tout en fournissant des données d'enquête contextuelles sur chaque OpsItem , associé et des ressources connexes. fournit OpsItems également des documents OpsCenter Automation que vous pouvez utiliser pour résoudre rapidement les problèmes. Systems Manager est intégré à OpsCenter .Amazon EventBridge Cela signifie que vous pouvez créer des règles EventBridge qui créent automatiquement des éléments OpsItems pour n'importe quel service AWS qui publie des événements sur EventBridge. Pour plus d'informations, consultez la section AWS Systems Manager OpsCenter (p. 176).

Amazon Simple Notification Service

Vous pouvez configurer Amazon Simple Notification Service (Amazon SNS) pour envoyer des notifications sur le statut des commandes que vous envoyez à l'aide de la fonctionnalité Systems Manager Fonctionnalité Exécuter la commande ou des Fenêtres de maintenance. Amazon SNS coordonne et gère l'envoi et la remise de notifications aux clients ou aux points de terminaison qui sont abonnés à des rubriques Amazon SNS. Vous pouvez recevoir une notification chaque fois qu'une

commande change de statut ou passe à un statut spécifique, par exemple, Échec ou Expiré. Lorsque vous envoyez une commande à plusieurs instances, vous pouvez recevoir une notification pour chaque copie de la commande envoyée à une instance spécifique. Pour plus d'informations, consultez la section Surveillance des changements de statut Systems Manager à l'aide de notifications Amazon SNS (p. 1344).

AWS Trusted Advisor et Personal Health Dashboard

Trusted Advisor tire profit des bonnes pratiques acquises à travers la satisfaction de centaines de milliers de clients AWS. Trusted Advisor examine votre environnement AWS, puis effectue des recommandations lorsqu'il est possible de faire des économies, d'améliorer la disponibilité et les performances du système, ou de remédier à des failles de sécurité. Tous les clients AWS ont accès à cinq contrôles Trusted Advisor. Les clients avec un plan de support Business ou Enterprise peuvent afficher tous les contrôles Trusted Advisor. Pour plus d'informations, consultez la section Tableaux de bord Personal Health et Trusted Advisor hébergés par Systems Manager (p. 210).

Contenu connexe

- Surveillance des AWS Systems Manager (p. 1326)

Validation de la conformité pour le AWS Systems Manager

Cette rubrique traite AWS Systems Manager de la conformité avec les programmes d'assurance tiers. Pour plus d'informations sur l'affichage des données de conformité pour vos instances gérées, consultez Conformité de la configuration AWS Systems Manager (p. 778) .

Des auditeurs tiers évaluent la sécurité et la conformité d'AWS Systems Manager dans le cadre de plusieurs programmes de conformité AWS. Il s'agit notamment des certifications SOC, PCI, FedRAMP , HIPAA et autres.

Pour obtenir la liste des services AWS relevant de programmes de conformité spécifiques, consultez les Services AWS relevant de programmes de conformité. Pour obtenir des renseignements généraux, consultez les Programmes de conformitéAWS .

Vous pouvez télécharger les rapports de l'audit externe avec AWS Artifact. Pour plus d'informations, consultez Téléchargement de rapports dans AWS Artifact.

Votre responsabilité en matière de conformité lorsque vous utilisez le Systems Manager est déterminée par la sensibilité de vos données, des objectifs de conformité de votre entreprise, ainsi que de la législation et de la réglementation en vigueur. AWS fournit les ressources suivantes pour faciliter le respect de la conformité :

- Guides de démarrage rapide de la sécurité et de la conformité . Ces guides de déploiement proposent des considérations architecturales et fournissent des étapes pour déployer des environnements de référence centrés sur la sécurité et la conformité sur AWS.
- Livre blanc Création d'une architecture pour la sécurité et la conformité HIPAA. Ce livre blanc décrit comment les entreprises peuvent utiliser AWS pour créer des applications conformes à la loi HIPAA.
- Ressources de conformité AWS – Cet ensemble de manuels et de guides peut s'appliquer à votre secteur et à votre emplacement.
- Évaluation des ressources à l'aide de règles dans le Manuel du développeur AWS Config. Le service AWS Config évalue dans quelle mesure vos configurations de ressources sont conformes aux pratiques internes, aux directives sectorielles et aux réglementations.
- AWS Security Hub. Ce service AWS fournit une vue complète de votre état de sécurité au sein d'AWS qui vous permet de vérifier votre conformité aux normes sectorielles et aux bonnes pratiques de sécurité.

Résilience dans AWS Systems Manager

L'infrastructure AWS mondiale repose sur des AWS régions et des zones de disponibilité.AWS Les régions fournissent plusieurs zones de disponibilité physiquement séparées et isolées, reliées par un réseau à latence faible, à débit élevé et à forte redondance. 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 plus hautement disponibles, tolérantes aux pannes et évolutives que les infrastructures traditionnelles à un ou plusieurs centres de données.

Pour de plus amples d'informations sur les régions et les zones de disponibilité AWS, veuillez consulter Infrastructure mondiale d'AWS .

Sécurité de l'infrastructure dans AWS Systems Manager

En tant que service géré, AWS Systems Manager est protégé par les procédures de sécurité du réseau AWS mondial qui sont décrites dans :Amazon Web Services Livre blanc Présentation des processus de sécurité .

Vous utilisez les appels d'API publiés dans AWS pour accéder à Systems Manager via le réseau. Les clients doivent prendre en charge le protocole TLS (Transport Layer Security) 1.0 ou version ultérieure. Nous recommandons TLS 1.2 ou version ultérieure. 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 telles que Java 7 et 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 mandataire IAM. Vous pouvez également utiliser AWS Security Token Service (AWS STS) pour générer des informations d'identification de sécurité temporaires et signer les demandes.

Configuration et analyse des vulnérabilités dans AWS Systems Manager

AWS gère les tâches de sécurité de base comme les correctifs du système d'exploitation invité et de base de données, la configuration du pare-feu, et la reprise après sinistre. Ces procédures ont été vérifiées et certifiées par les tiers appropriés. Pour plus de détails, consultez les ressources suivantes.

- Validation de la conformité pour le AWS Systems Manager (p. 1320)
- Modèle de responsabilité partagée
- Amazon Web Services : Présentation des processus de sécurité (livre blanc)

Bonnes pratiques de sécurité pour Systems Manager

Systems Manager fournit différentes fonctions de sécurité à prendre en compte lorsque vous développez et implémentez vos propres stratégies de sécurité. Les bonnes pratiques suivantes doivent être considérées comme des instructions générales et ne représentent pas une solution de sécurité complète. Étant donné

que ces bonnes pratiques peuvent ne pas être appropriées ou suffisantes pour votre environnement, considérez-les comme des remarques utiles plutôt que comme des recommandations.

Rubriques

- Bonnes pratiques de sécurité préventive pour Systems Manager (p. 1322)
- Bonnes pratiques de surveillance et d'audit pour Systems Manager (p. 1324)

Bonnes pratiques de sécurité préventive pour Systems Manager

Les bonnes pratiques suivantes pour Systems Manager peuvent aider à éviter les incidents de sécurité.

Implémentation d'un accès sur la base du moindre privilège

Lorsque vous accordez des autorisations, vous choisissez qui obtient les autorisations pour telles ou telles ressources Systems Manager. Vous activez des actions spécifiques que vous souhaitez autoriser sur ces ressources. Par conséquent, vous devez accorder uniquement les autorisations qui sont requises pour exécuter une tâche. L'implémentation d'un accès sur la base du moindre privilège est fondamentale pour réduire les risques en matière de sécurité et l'impact que pourraient avoir des d'erreurs ou des actes de malveillance.

Les outils suivants sont disponibles pour l'implémentation d'un accès sur la base du moindre privilège :

- Stratégies utilisateur IAM et Limites d'autorisations pour des entités IAM
- Stratégies de contrôle de service

Utiliser SecureString des paramètres pour chiffrer et protéger des données de secret

Un paramètre `SecureString` correspond à des données sensibles qui doivent être stockées et référencées de manière sécurisée. Si vous avez des données que vous ne voulez pas que les utilisateurs modifient ou référencent en texte brut, comme des mots de passe ou des clés de licence, créez ces paramètres à l'aide du type de données `SecureString`. Parameter Store utilise une clé principale client (CMK) AWS Key Management Service (AWS KMS) pour chiffrer la valeur du paramètre. AWS KMS utilise soit une clé CMK gérée par le client, soit une clé CMK gérée par AWS lors du chiffrement de la valeur du paramètre. Pour une sécurité maximale, nous vous recommandons d'utiliser votre propre clé CMK. Si vous utilisez la clé CMK gérée par AWS par défaut, tout utilisateur autorisé à exécuter les actions GetParameter et GetParameters dans votre compte peut afficher ou récupérer le contenu de tous les paramètres `SecureString`. Si vous utilisez le chiffrement géré par le client CMKs pour chiffrer vos `SecureString` valeurs sécurisées, vous pouvez utiliser IAM des stratégies et des stratégies de clé pour gérer les autorisations de chiffrement et de déchiffrement des paramètres. Vous ne pouvez pas établir de stratégies de contrôle d'accès pour ces opérations lorsque vous utilisez la clé CMK gérée par AWS par défaut.

Pour plus d'informations, consultez Restriction de l'accès aux paramètres Systems Manager à l'aide des stratégies IAM (p. 223) et Comment AWS Systems Manager Parameter Store utilise AWS KMS dans le AWS Key Management Service Developer Guide.

Définir allowedValues et allowedPattern pour les paramètres de document

Vous pouvez valider l'entrée utilisateur pour les paramètres SSM du document en définissant `allowedValues` et `allowedPattern` . Pour `allowedValues` , vous définissez un tableau de valeurs autorisées pour le paramètre. Si un utilisateur saisit une valeur non autorisée, l'exécution échoue. Pour `allowedPattern`, vous définissez une expression régulière qui valide si l'entrée utilisateur correspond au modèle défini pour le paramètre. Si l'entrée utilisateur ne correspond pas au modèle autorisé, l'exécution échoue.

Pour plus d'informations sur `allowedValues` et `allowedPattern`, consultez Syntaxe du document SSM (p. 1226).

Utilisation d'un Amazon Virtual Private Cloud (Amazon VPC) et de points de terminaison d'un VPC

Vous pouvez utiliser Amazon VPC pour lancer des ressources AWS dans un réseau virtuel que vous avez défini. Ce réseau virtuel ressemble beaucoup à un réseau traditionnel que vous pourriez exécuter dans votre propre data center, et présente l'avantage d'utiliser l'infrastructure évolutive d'AWS.

En implémentant un point de terminaison de VPC, vous pouvez connecter de manière privée votre VPC aux services AWS pris en charge et aux services de point de terminaison de VPC alimentés par AWS PrivateLink sans avoir besoin d'une passerelle Internet, d'un périphérique NAT, d'une connexion VPN ou dAWS Direct Connect'une connexion. Les instances de votre VPC ne requièrent pas d'adresses IP publiques pour communiquer avec les ressources du service. Le trafic entre votre VPC et les autres services ne quitte pas le réseau Amazon.

Pour plus d'informations sur la sécurité Amazon VPC, consultez (Facultatif) Créer un point de terminaison de VPC (p. 42) et Sécurité dans le Amazon VPC Guide de l'utilisateur.

Limiter Session Manager les utilisateurs aux sessions à l'aide de commandes interactives

AWS Systems Manager Session Manager fournit plusieurs méthodes pour démarrer des sessions (p. 924) sur vos instances gérées. Pour les connexions les plus sécurisées, vous pouvez exiger que les utilisateurs se connectent à l'aide de la méthode des commandes interactives afin de limiter l'interaction de l'utilisateur à une commande ou une séquence de commandes spécifique. Cela vous permet de gérer les actions interactives qu'un utilisateur peut effectuer. Pour plus d'informations, consultez Démarrage d'une session (commandes interactives) (p. 927).

Fournir des autorisations d'instance temporaires pour les exécutions d'Automation

Lors d'une exécution d'Automation, vos instances peuvent avoir besoin d'autorisations nécessaires pour cette exécution uniquement, mais pas pour d'autres opérations Systems Manager. Par exemple, un flux de travail Automation peut nécessiter une instance pour appeler une action d'API particulière ou accéder à une ressource AWS spécifiquement pendant le flux de travail. Si ces appels ou ressources sont ceux auxquels vous souhaitez limiter l'accès, vous pouvez fournir des autorisations supplémentaires temporaires pour vos instances dans le document Automation lui-même au lieu d'ajouter les autorisations à votre profil d'instance IAM. À la fin de l'exécution d'Automation, les autorisations temporaires sont supprimées. Pour plus d'informations, consultez l'article relatif à l'affectation d'autorisations d'instance temporaires avec AWS Systems Manager Automation sur le blog AWS consacré à la gestion et la gouvernance.

Gardez AWS les outils et à jourSystems Manager

AWS publie régulièrement des versions mises à jour des outils et plug-ins que vous pouvez utiliser dans vos opérations AWS et Systems Manager. La mise à jour de ces ressources garantit que les utilisateurs et les instances de votre compte ont accès aux fonctionnalités et aux fonctions de sécurité les plus récentes de ces outils.

- Agent SSM – AWS Systems Manager Agent (Agent SSM) est un logiciel Amazon qui peut être installé et configuré sur une instance EC2, un serveur sur site ou une machine virtuelle (VM). Agent SSM permet à Systems Manager de mettre à jour, de gérer et de configurer ces ressources. Nous vous recommandons de rechercher les nouvelles versions, ou d'automatiser les mises à jour de l'agent, au moins toutes les deux semaines. Pour plus d'informations, consultez la section Automatisation des mises à jour versAgent SSM (p. 109).
- AWS CLI – L'AWS Command Line Interface (AWS CLI) est un outil à code source libre qui vous permet d'interagir avec les services AWS à l'aide des commandes de votre shell de ligne de commande. Pour mettre à jour l'AWS CLI, vous exécutez la même commande que pour installer l'AWS CLI. Nous vous recommandons de créer une tâche planifiée sur votre ordinateur local pour exécuter la commande appropriée pour votre système d'exploitation au moins une fois toutes les deux semaines. Pour plus d'informations sur les commandes d'installation, consultez Installation de l'AWS Command Line Interface dans le AWS Command Line Interface Guide de l'utilisateur.
- Outils AWS pour Windows PowerShell–Les Outils pour Windows PowerShell sont un ensemble de PowerShell modules qui reposent sur les fonctionnalités exposées par le kit AWS SDK pour .NET. Les vous Outils AWS pour Windows PowerShell permettent d'écrire le script d'opérations

sur vos ressources AWS à partir de la ligne de PowerShell commande. Vous devez mettre à jour périodiquement la version des Outils AWS pour Windows PowerShell que vous exécutez localement, à mesure que des versions mises à jour sont publiées. Pour plus d'informations, consultez Mise à jour de l' Outils AWS pour Windows PowerShell sous Windows ou Mise à jour de l' Outils AWS pour Windows PowerShell sous Linux ou macOS .

- Session Manager plugin – Si les utilisateurs de votre organisation disposant des autorisations nécessaires pour utiliser Session Manager veulent se connecter à une instance à l'aide de l'AWS CLI, ils doivent d'abord installer le Session Manager plugin sur leurs ordinateurs locaux. Pour mettre à jour le plug-in, vous exécutez la même commande que pour l'installer. Nous vous recommandons de créer une tâche planifiée sur votre ordinateur local pour exécuter la commande appropriée pour votre système d'exploitation au moins une fois toutes les deux semaines. Pour plus d'informations, consultez la section (Facultatif) Installer le Session Manager plugin pour l'AWS CLI (p. 917).
- Agent CloudWatch – Vous pouvez configurer et utiliser l'Agent CloudWatch pour collecter des métriques et des journaux à partir de vos instances EC2, instances sur site et machines virtuelles (VM). Ces journaux peuvent être envoyés à Amazon CloudWatch Logs à des fins de surveillance et d'analyse. Nous vous recommandons de rechercher les nouvelles versions, ou d'automatiser les mises à jour de l'agent, au moins toutes les deux semaines. Pour les mises à jour les plus simples, utilisez la configuration rapide AWS Systems Manager. Pour plus d'informations, consultez la section AWS Systems Manager Configuration rapide (p. 17).

Bonnes pratiques de surveillance et d'audit pour Systems Manager

Les bonnes pratiques suivantes pour Systems Manager peuvent aider à détecter les vulnérabilités et les incidents de sécurité potentiels.

Identifier et auditer toutes vos Systems Manager ressources

L'identification de vos ressources informatiques est un aspect crucial de la gouvernance et de la sécurité. Vous devez identifier toutes vos ressources Systems Manager pour évaluer leur niveau de sécurité et agir sur les zones de vulnérabilité potentielles.

Utilisez Tag Editor pour identifier les ressources sensibles en termes de sécurité ou d'audit, puis utilisez les balises générées lorsque vous devez rechercher ces ressources. Pour plus d'informations, consultez Rechercher des ressources à baliser dans le Guide de l'utilisateur Groupes de ressources AWS.

Créez des groupes de ressources pour vos ressourcesSystems Manager. Pour plus d'informations, consultez Groupes de ressources AWS.

Implémentation de la surveillance à l'aide des outils Amazon CloudWatch de surveillance

La surveillance constitue une part importante de la gestion de la fiabilité, de la disponibilité et des performances de vos instances Systems Manager et de vos solutions AWS. CloudWatch fournit différents outils et services que vous pouvez utiliser pour surveiller Systems Manager et vos autres services AWS. Pour de plus amples informations, veuillez consulter Envoi de journaux d'instance à CloudWatch Logs (Agent CloudWatch) (p. 1327) et Surveillance des événements Systems Manager avec Amazon EventBridge (p. 1340).

Utiliser AWS CloudTrail

AWS CloudTrail fournit un enregistrement des actions réalisées par un utilisateur, un rôle ou un service AWS dans Systems Manager. À l'aide des informations collectées par AWS CloudTrail, vous pouvez déterminer la demande qui a été envoyée à Systems Manager, l'adresse IP source à partir de laquelle la demande a été effectuée, l'auteur de la demande et la date de la demande, ainsi que d'autres informations. Pour plus d'informations, consultez la section Journalisation des appels d'API AWS Systems Manager avec AWS CloudTrail (p. 1336).

Activer AWS Config

AWS Config vous permet d'évaluer et d'auditer les configurations de vos ressources AWS. AWS Config surveille les configurations de ressources, ce qui vous permet d'évaluer les configurations enregistrées par rapport aux configurations sécurisées souhaitées. Avec AWS Config, vous pouvez examiner l'évolution des configurations et des relations entre les ressources AWS, consulter des historiques de configuration des ressources détaillés et déterminer votre conformité globale aux configurations spécifiées dans vos directives internes. Cela peut vous aider à simplifier le contrôle de la conformité, l'analyse de la sécurité, la gestion des modifications et le diagnostic de défaillances opérationnelles. Pour plus d'informations, consultez Configuration de AWS Config à l'aide de la console dans le Manuel du développeur AWS Config. Lors de la spécification des types de ressource à enregistrer, assurez-vous d'inclure les ressources Systems Manager.

Surveillance des conseils AWS de sécurité

Vous devez vérifier régulièrement les conseils de sécurité publiés dans Trusted Advisor pour votre compte AWS. Vous pouvez le faire par programmation à l'aide de describe-trusted-advisor-checks.

En outre, surveillez activement votre adresse e-mail principale enregistrée dans chacun de vos comptes AWS. AWS vous contactera à l'aide de cette adresse e-mail, en cas de problèmes de sécurité qui peuvent vous concerner.

Les problèmes opérationnels AWS avec des répercussions majeures sont publiés dans le tableau de bord de l'état des services AWS. Les problèmes opérationnels sont également publiés dans les différents comptes via le tableau de bord d'état personnel. Pour plus d'informations, consultez la documentation AWS Health.

Contenu connexe

- Amazon Web Services : Présentation des processus de sécurité (livre blanc)
- Mise en route : Suivez les bonnes pratiques de sécurité lorsque vous configurez vos ressources AWS (blog sur la sécurité AWS)
- Bonnes pratiques IAM
- Bonnes pratiques de sécurité dans AWS CloudTrail
- Bonnes pratiques de sécurité pour Amazon S3

Surveillance des AWS Systems Manager

La surveillance est un enjeu important pour assurer la fiabilité, la disponibilité et les performances de Systems Manager et de vos solutions AWS. Vous devez recueillir les données de surveillance de toutes les parties de votre solution AWS afin que vous puissiez déboguer plus facilement une éventuelle défaillance multipoint. Avant de commencer à surveiller Systems Manager, vous devez créer un plan de surveillance qui contient les réponses aux questions suivantes :

- Quels sont les objectifs de la surveillance ?
- Quelles sont les ressources à surveiller ?
- A quelle fréquence les ressources doivent-elles être surveillées ?
- Quels outils de supervision utiliser ?
- Qui exécute les tâches de surveillance ?
- Qui doit être informé en cas de problème ?

Une fois que vous avez défini vos objectifs de surveillance et créé votre plan de surveillance, l'étape suivante consiste à définir une référence pour les performances normales d'Systems Manager dans votre environnement. Vous devez mesurer les performances d'Systems Manager à différents moments et sous différentes conditions de charge. Lorsque vous supervisez Systems Manager, stockez l'historique des données de surveillance que vous avez collectées. Vous pouvez comparer les performances actuelles d'Systems Manager à leurs données historiques pour vous aider à identifier les modèles de performances normales et les anomalies de performances, et à concevoir les méthodes destinées à les prendre en compte.

Par exemple, vous pouvez surveiller le succès ou l'échec d'opérations telles que les flux de travail Automation, l'application de références de correctifs, les événements de fenêtre de maintenance et la conformité de la configuration.

Vous pouvez également surveiller l'utilisation de l'UC, les I/O disque et l'utilisation du réseau de vos instances gérées. Lorsque les performances se trouvent en dehors de votre référence établie, il se peut que vous ayez besoin de reconfigurer l'instance ou de l'optimiser pour réduire l'utilisation de l'UC, améliorer les E/S disque ou réduire le trafic réseau. Pour de plus amples informations sur la surveillance de vos instances EC2, veuillez consulter Surveillance de Amazon EC2 dans le Amazon EC2 Guide de l'utilisateur pour les instances Linux.

Rubriques

- Outils de surveillance (p. 1327)
- Envoi de journaux d'instance à CloudWatch Logs (Agent CloudWatch) (p. 1327)
- Envoi de Agent SSM journaux àCloudWatch Logs (p. 1333)
- Surveillance des métriques Fonctionnalité Exécuter la commande avec Amazon CloudWatch (p. 1335)
- Journalisation des appels d'API AWS Systems Manager avec AWS CloudTrail (p. 1336)
- Configuration d'Amazon CloudWatch Logs pour Fonctionnalité Exécuter la commande (p. 1338)
- Surveillance des événements Systems Manager avec Amazon EventBridge (p. 1340)
- Surveillance des changements de statut Systems Manager à l'aide de notifications Amazon SNS (p. 1344)

Outils de surveillance

Le contenu de ce chapitre fournit des informations sur l'utilisation des outils disponibles pour surveiller vos ressources AWS Systems Manager et d'autres ressources AWS. Pour obtenir une liste d'outils plus complète, consultez Journalisation et surveillance dans AWS Systems Manager (p. 1318).

Envoi de journaux d'instance à CloudWatch Logs (Agent CloudWatch)

Vous pouvez configurer et utiliser l'Agent CloudWatch Amazon pour collecter des métriques et des journaux auprès de vos instances, au lieu d'utiliser l'Agent SSM pour ces tâches. L'Agent CloudWatch permet de recueillir plus de métriques sur les instances EC2 qu'avec l'Agent SSM. En outre, vous pouvez recueillir des métriques des serveurs sur site en installant l'Agent CloudWatch.

Vous pouvez également stocker les paramètres de configuration d'agent dans le Systems Manager Parameter Store, en vue de les utiliser avec l'Agent CloudWatch.

> **Note**
>
> Actuellement, AWS Systems Manager prend en charge la migration depuis l'Agent SSM vers l'Agent CloudWatch pour la collecte des journaux et des mesures sur les versions 64 bits de Windows uniquement. Pour de plus amples informations à propos de la configuration de l'Agent CloudWatch sur d'autres systèmes d'exploitation et de l'utilisation de l'Agent CloudWatch, veuillez consulter Collecte de métriques et de journaux à partir d'instances Amazon Elastic Compute Cloud et de serveurs sur site avec l'Agent CloudWatch dans le Guide de l'utilisateur Amazon CloudWatch.
> Vous pouvez utiliser l'Agent CloudWatch sur d'autres systèmes d'exploitation pris en charge, mais vous ne pourrez pas utiliser Systems Manager pour effectuer une migration d'outil.

L'Agent SSM écrit des informations relatives aux exécutions, actions planifiées, erreurs et états d'intégrité dans les fichiers journaux de chaque instance. La connexion manuelle à une instance pour afficher les fichiers journaux et résoudre un problème avec l'Agent SSM est une opération chronophage. Afin d'augmenter l'efficacité de la surveillance des instances, vous pouvez configurer l'Agent SSM lui-même ou l'Agent CloudWatch pour envoyer ces données de journaux à Amazon CloudWatch Logs.

> **Important**
>
> L'agent Agent CloudWatch unifié a remplacé Agent SSM comme outil d'envoi des données de journaux à Amazon CloudWatch Logs. La prise en charge d'Agent SSM pour envoyer les données de journaux sera bientôt obsolète. Nous vous recommandons d'utiliser l'agent Agent CloudWatch unifié pour vos processus de collecte de journal. Pour plus d'informations, consultez les rubriques suivantes :
>
> - Envoi de journaux d'instance à CloudWatch Logs (Agent CloudWatch) (p. 1327)
> - Migrer la collecte de journaux d'instances Windows Server vers l'Agent CloudWatch (p. 1328)
> - Collecte des métriques depuis les instances Amazon Elastic Compute Cloud et les serveurs locaux avec l'Agent CloudWatch dans le Guide de l'utilisateur Amazon CloudWatch

CloudWatch Logs permet de surveiller les données des journaux en temps réel, de rechercher et de filtrer ces données en créant un ou plusieurs filtres de métriques et d'archiver et de récupérer les données d'historique lorsque vous en avez besoin. Pour de plus amples informations sur CloudWatch Logs, veuillez consulter Amazon CloudWatch Logs User Guide.

La configuration d'un agent pour envoyer des données de journaux à Amazon CloudWatch Logs offre les avantages suivants :

- Stockage centralisé des fichiers journaux pour l'ensemble de vos fichiers journaux de l'Agent SSM.
- Accès plus rapide aux fichiers pour examiner les erreurs.
- Conservation des fichiers journaux illimitée (configurable).
- Les journaux peuvent être conservés et accessibles quel que soit le statut de l'instance.
- Accès à d'autres fonctionnalités CloudWatch telles que les métriques et les alarmes.

Pour de plus amples informations sur la surveillance de l'activité Session Manager, veuillez consulter Audit et journalisation de l'activité de session (p. 929).

Migrer la collecte de journaux d'instances Windows Server vers l'Agent CloudWatch

Si vous utilisez actuellement l'Agent SSM sur des instances Windows Server prises en charge pour envoyer des fichiers journaux de l'Agent SSM à Amazon CloudWatch Logs, vous pouvez utiliser Systems Manager comme outil de collecte de journaux pour migrer des données de l'Agent SSM vers l'Agent CloudWatch et migrer vos paramètres de configuration.

LAgent CloudWatch' n'est pas pris en charge sur les versions 32 bits d' Windows Server .

Pour les instances EC2 64 bits pour Windows Server, vous pouvez effectuer la migration vers l'Agent CloudWatch automatiquement ou manuellement. Pour les serveurs sur site et les machines virtuelles sur site, le processus doit être réalisé manuellement.

> **Note**
>
> Au cours du processus de migration, les données envoyées à CloudWatch peuvent être interrompues ou dupliquées. Vos métriques et données de journaux sont enregistrées précisément à nouveau dans CloudWatch une fois la migration terminée.

Nous recommandons de tester la migration sur un nombre limité d'instances avant de migrer une flotte entière vers l'Agent CloudWatch. Après la migration, si vous préférez réaliser la collecte de journaux avec l'Agent SSM, vous pouvez reprendre son utilisation.

> **Important**
>
> Dans les cas suivants, vous ne pourrez pas effectuer la migration vers l'Agent CloudWatch en appliquant la procédure décrite dans cette rubrique :
>
> - La configuration existante pour l'Agent SSM spécifie plusieurs régions.
> - La configuration existante pour l'Agent SSM spécifie plusieurs jeux d'informations d'identification pour l'accès/la clé secrète.
>
> Dans ces cas, il est nécessaire de désactiver la collecte de journaux dans l'Agent SSM et d'installer l'Agent CloudWatch sans processus de migration. Pour de plus amples informations, consultez les rubriques suivantes :
>
> - Installer l'CloudWatchagent sur une instance EC2
> - Installer l'CloudWatchagent sur un serveur sur site

Avant de commencer

Avant de lancer une migration vers l'Agent CloudWatch pour la collecte de journaux, vérifiez que les instances sur lesquelles vous allez effectuer la migration remplissent les conditions suivantes :

- Le système d'exploitation est une version 64 bits de Windows Server.
- L'Agent SSM 2.2.93.0 ou une version ultérieure est installé sur l'instance.
- Agent SSM est configuré pour la surveillance sur l'instance.

Rubriques

- Migration automatique vers l'Agent CloudWatch (p. 1329)
- Migration manuelle vers l'Agent CloudWatch (p. 1330)

Migration automatique vers l'Agent CloudWatch

Pour les instances EC2 pour Windows Server uniquement, vous pouvez utiliser la console AWS Systems Manager ou l'AWS CLI pour effectuer automatiquement la migration vers l'Agent CloudWatch et l'adopter comme outil de collecte de journaux.

> **Note**
>
> Actuellement, AWS Systems Manager prend en charge la migration depuis l'Agent SSM vers l'Agent CloudWatch pour la collecte des journaux et des mesures sur les versions 64 bits de Windows uniquement. Pour de plus amples informations à propos de la configuration de l'Agent CloudWatch sur d'autres systèmes d'exploitation et de l'utilisation de l'Agent CloudWatch, veuillez consulter Collecte de métriques et de journaux à partir d'instances Amazon Elastic Compute Cloud et de serveurs sur site avec l'Agent CloudWatch dans le Guide de l'utilisateur Amazon CloudWatch.
> Vous pouvez utiliser l'Agent CloudWatch sur d'autres systèmes d'exploitation pris en charge, mais vous ne pourrez pas utiliser Systems Manager pour effectuer une migration d'outil.

Une fois la migration terminée, vérifiez les résultats dans CloudWatch afin de vous assurer que vous recevez les métriques, journaux ou journaux d'événements Windows prévus. Si vous êtes satisfait des résultats, vous pouvez éventuellement Stocker les paramètres de configuration de l'Agent CloudWatch dans Parameter Store (p. 1332). Si la migration n'aboutit pas ou si les résultats ne sont pas ceux attendus, vous pouvez Retourner à la collecte de journaux avec l'Agent SSM (p. 1332).

> **Note**
>
> Si vous souhaitez migrer un fichier de configuration source incluant une entrée `{hostname}`, sachez que l'entrée `{hostname}` peut changer la valeur du champ une fois la migration terminée. Par exemple, supposons que l'`"LogStream": "{hostname}"`entrée suivante correspond à un serveur nommé MyLogServer001 .

```
{
"Id": "CloudWatchIISLogs",
"FullName":
 "AWS.EC2.Windows.CloudWatch.CloudWatchLogsOutput,AWS.EC2.Windows.CloudWatch",
"Parameters": {
"AccessKey": "",
"SecretKey": "",
"Region": "us-east-1",
"LogGroup": "Production-Windows-IIS",
"LogStream": "{hostname}"
     }
}
```

> Après la migration, cette entrée sera mappée à un domaine, tel que ip-11-1-1-11.production.ExampleCompany.com. Pour conserver la valeur du nom d'hôte local, spécifiez `{local_hostname}` au lieu de `{hostname}`.

Pour effectuer une migration automatique vers l'Agent CloudWatch (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fonctionnalité Exécuter la commande, puis Run command (Exécuter la commande).
3. Dans la liste Command document (Document de commande), choisissez `AmazonCloudWatch-MigrateCloudWatchAgent`.
4. Dans la section Cibles, choisissez une option et sélectionnez les instances à mettre à jour.
5. Cliquez sur Exécuter.

Pour effectuer automatiquement la migration vers l'Agent CloudWatch (AWS CLI)

- Exécutez la commande suivante :

```
aws ssm send-command --document-name AmazonCloudWatch-MigrateCloudWatchAgent --targets
 Key=instanceids,Values=ID1,ID2,ID3
```

ID1, *ID2*, et *ID3* représente l' IDs des instances que vous souhaitez mettre à jour, par exemple i-02573cafcfEXAMPLE .

Migration manuelle vers l'Agent CloudWatch

Pour les instances Windows Server sur site ou les instances EC2 pour Windows Server, suivez la procédure suivante pour migrer manuellement la collecte de journaux vers l'Agent CloudWatch.

Note

Si vous souhaitez migrer un fichier de configuration source incluant une entrée `{hostname}`, sachez que l'entrée `{hostname}` peut changer la valeur du champ une fois la migration terminée. Par exemple, supposons que l'`"LogStream": "{hostname}"`entrée suivante correspond à un serveur nommé MyLogServer001 .

```
{
"Id": "CloudWatchIISLogs",
"FullName":
 "AWS.EC2.Windows.CloudWatch.CloudWatchLogsOutput,AWS.EC2.Windows.CloudWatch",
"Parameters": {
"AccessKey": "",
"SecretKey": "",
"Region": "us-east-1",
"LogGroup": "Production-Windows-IIS",
"LogStream": "{hostname}"
     }
}
```

Après la migration, cette entrée sera mappée à un domaine, tel que ip-11-1-1-11.production.ExampleCompany.com. Pour conserver la valeur du nom d'hôte local, spécifiez `{local_hostname}` au lieu de `{hostname}`.

Un : Pour installer l'CloudWatchagent (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fonctionnalité Exécuter la commande, puis Run command (Exécuter la commande).

3. Dans la liste Command document (Document de commande), choisissez `AWS-ConfigureAWSPackage`.
4. Dans la section Cibles, choisissez une option et sélectionnez les instances à mettre à jour.
5. Dans la liste Action, choisissez `Install`.
6. Dans Nom, tapez **`AmazonCloudWatchAgent`**.
7. Dans Version, tapez `latest` si elle n'est pas déjà fournie par défaut.
8. Cliquez sur Exécuter.

Deux : Pour mettre à jour le format JSON des données de configuration

- Pour mettre à jour le formatage JSON des paramètres de configuration existants pour l' Agent CloudWatch , utilisez l'instance AWS Systems Manager Fonctionnalité Exécuter la commande ou connectez-vous directement à l'instance avec une connexion RDP pour exécuter les PowerShell commandes Windows suivantes sur l'instance, l'une après l'autre :

```
cd ${Env:ProgramFiles}\\Amazon\\AmazonCloudWatchAgent
```

```
.\\amazon-cloudwatch-agent-config-wizard.exe --isNonInteractiveWindowsMigration
```

  `{Env:ProgramFiles}` représente l'emplacement où se trouve le dossier Amazon contenant l' Agent CloudWatch , généralement C:\\Program Files.

Trois : Pour configurer et démarrer l'CloudWatchagent (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fonctionnalité Exécuter la commande, puis Run command (Exécuter la commande).
3. Dans la liste Command document (Document de commande), choisissez `AWS-RunPowerShellScript`.
4. Dans la section Cibles, choisissez une option et sélectionnez les instances à mettre à jour.
5. Dans la zone Commands, entrez les deux commandes suivantes :

```
cd ${Env:ProgramFiles}\Amazon\AmazonCloudWatchAgent
```

```
.\amazon-cloudwatch-agent-ctl.ps1 -a fetch-config -m ec2 -c file:config.json -s
```

   `{Env:ProgramFiles}` représente l'emplacement où se trouve le dossier Amazon contenant l' , généralement Agent CloudWatch C:\\Program Files.
6. Cliquez sur Exécuter.

Quatre : Pour désactiver la collecte de journaux dans Agent SSM (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fonctionnalité Exécuter la commande, puis Run command (Exécuter la commande).
3. Dans la liste Command document (Document de commande), choisissez `AWS-ConfigurecloudWatch`.

4. Dans la section Cibles, choisissez une option et sélectionnez les instances à mettre à jour.
5. Dans la liste des statuts, choisissez `Disabled`.
6. Cliquez sur Exécuter.

   Une fois la procédure terminée, vérifiez vos journaux dans CloudWatch afin de vous assurer que vous recevez les métriques, journaux ou journaux d'événements Windows prévus. Si vous êtes satisfait des résultats, vous pouvez éventuellement Stocker les paramètres de configuration de l'Agent CloudWatch dans Parameter Store (p. 1332). Si la migration n'aboutit pas ou si les résultats ne sont pas ceux attendus, vous pouvez Retourner à la collecte de journaux avec l'Agent SSM (p. 1332).

Stocker les paramètres de configuration de l'Agent CloudWatch dans Parameter Store

Vous pouvez stocker le contenu du fichier de configuration de l'Agent CloudWatch dans Parameter Store. Si vous conservez ces données de configuration dans un paramètre, plusieurs instances peuvent en tirer leurs paramètres de configuration, ce qui vous évite de créer ou de mettre à jour manuellement les fichiers de configuration sur vos instances. Vous pouvez, par exemple, utiliser Fonctionnalité Exécuter la commande pour écrire le contenu du paramètre dans des fichiers de configuration sur plusieurs instances ou utiliser State Manager pour éviter des écarts dans les paramètres de configuration de l'Agent CloudWatch dans un parc d'instances.

Lorsque vous exécutez l'assistant de configuration de l'Agent CloudWatch, vous pouvez décider de laisser l'assistant enregistrer vos paramètres de configuration en tant que nouveau paramètre dans Parameter Store. Pour de plus amples informations sur l'exécution de l'assistant de configuration de l'Agent CloudWatch, veuillez consulter Création du fichier de configuration de l'Agent CloudWatch avec l'assistant.

Si vous avez exécuté l'assistant sans choisir l'option d'enregistrement des paramètres sous la forme d'un seul paramètre ou si vous avez créé manuellement le fichier de configuration de l'Agent CloudWatch, vous pouvez récupérer les données à enregistrer dans un seul paramètre sur votre instance dans le fichier suivant :

```
${Env:ProgramFiles}\Amazon\AmazonCloudWatchAgent\config.json
```

`{Env:ProgramFiles}` représente l'emplacement où se trouve le dossier Amazon contenant l' , généralement Agent CloudWatch C:\\Program Files.

Nous vous conseillons de conserver une copie de sauvegarde du code JSON contenu dans ce fichier dans un emplacement autre que l'instance même.

Pour plus d'informations sur la création d'un paramètre, consultez Création de paramètres Systems Manager (p. 243).

Pour de plus amples informations sur l'Agent CloudWatch, veuillez consulter Collecte de métriques à partir d'instances Amazon Elastic Compute Cloud et de serveurs sur site avec l'agent CloudWatch dans le Guide de l'utilisateur Amazon CloudWatch.

Retourner à la collecte de journaux avec l'Agent SSM

Si vous souhaitez utiliser à nouveau l'Agent SSM pour collecter les journaux, suivez les étapes décrites dans cette section.

Un : Pour récupérer les données de configuration à partir deAgent SSM

1. Sur l'instance où vous souhaitez recommencer à collecter les journaux avec l'Agent SSM, localisez le contenu du fichier de configuration de l'Agent SSM. Ce fichier JSON se trouve généralement à l'emplacement suivant :

```
${Env:ProgramFiles}\\Amazon\\SSM\\Plugins\\awsCloudWatch\
\AWS.EC2.Windows.CloudWatch.json
```

{Env:ProgramFiles} représente l'emplacement où se trouve le `Amazon` dossier, généralement C:\ \Program Files.

2. Copiez ces données dans un fichier texte pour les utiliser à une étape ultérieure.

   Nous vous conseillons de stocker une copie de sauvegarde du fichier JSON dans un emplacement autre que l'instance elle-même.

Deux : Pour désinstaller l' Agent CloudWatch (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fonctionnalité Exécuter la commande, puis Run command (Exécuter la commande).
3. Dans la liste Command document (Document de commande), choisissez `AWS-ConfigureAWSPackage`.
4. Dans la section Cibles, choisissez une option et sélectionnez les instances à mettre à jour.
5. Dans la liste Action, choisissez `Uninstall`.
6. Dans Nom, tapez **`AmazonCloudWatchAgent`**.
7. Cliquez sur Exécuter.

Trois : Pour réactiver la collecte de journaux dans Agent SSM (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fonctionnalité Exécuter la commande, puis Run command (Exécuter la commande).
3. Dans la liste Command document (Document de commande), choisissez `AWS-ConfigureCloudWatch`.
4. Dans la section Cibles, choisissez une option et sélectionnez les instances à mettre à jour.
5. Dans la liste des statuts, choisissez `Enabled`.
6. Dans la zone Propriétés collez le contenu des anciennes données de configuration que vous avez enregistrées dans le fichier texte.
7. Cliquez sur Exécuter.

Envoi de Agent SSM journaux àCloudWatch Logs

L'agent AWS Systems Manager (Agent SSM) est un logiciel Amazon qui s'exécute sur vos instances EC2 et vos instances hybrides (instances sur site et machines virtuelles) configurées pour Systems Manager. L'Agent SSM traite les demandes provenant du service Systems Manager dans le cloud et configure votre machine comme spécifié dans la demande. Pour plus d'informations sur Agent SSM, consultez Utilisation des Agent SSM (p. 72).

En outre, en suivant la procédure ci-après, vous pouvez configurer l'Agent SSM pour qu'il envoie les données des journaux à Amazon CloudWatch Logs.

Avant de commencer

Créez un groupe de journaux dans Amazon CloudWatch Logs. Pour de plus amples informations, veuillez consulter Créer un groupe de journaux dans CloudWatch Logs dans le Amazon CloudWatch Logs User Guide.

Pour configurer l'Agent SSM afin d'envoyer les journaux à CloudWatch

1. Connectez-vous à une instance et recherchez le fichier suivant :

   Linux

   Sur la plupart des types d'instances Linux : `/etc/amazon/ssm/seelog.xml.template`.

   Sur Ubuntu Server 20.10 STR & 20.04, 18.04 et 16.04 LTS : `/snap/amazon-ssm-agent/current/seelog.xml.template`.

   Windows

   `%ProgramFiles%\Amazon\SSM\seelog.xml.template`
2. Remplacez le nom du fichier `seelog.xml.template` par `seelog.xml`.
3. Ouvrez le fichier `seelog.xml` dans un éditeur de texte, puis localisez la section ci-après.

   Linux

   ```
   <outputs formatid="fmtinfo">
      <console formatid="fmtinfo"/>
      <rollingfile type="size" filename="/var/log/amazon/ssm/amazon-ssm-agent.log"
    maxsize="30000000" maxrolls="5"/>
      <filter levels="error,critical" formatid="fmterror">
         <rollingfile type="size" filename="/var/log/amazon/ssm/errors.log"
    maxsize="10000000" maxrolls="5"/>
      </filter>
   </outputs>
   ```

   Windows

   ```
   <outputs formatid="fmtinfo">
      <console formatid="fmtinfo"/>
      <rollingfile type="size" maxrolls="5" maxsize="30000000"
    filename="{{LOCALAPPDATA}}\Amazon\SSM\Logs\amazon-ssm-agent.log"/>
      <filter formatid="fmterror" levels="error,critical">
         <rollingfile type="size" maxrolls="5" maxsize="10000000"
    filename="{{LOCALAPPDATA}}\Amazon\SSM\Logs\errors.log"/>
      </filter>
   </outputs>
   ```
4. Modifiez le fichier et ajoutez un élément de nom personnalisé après la balise de fermeture </filter>. Dans l'exemple suivant, le nom personnalisé tel qu'il a été spécifié en tant que `cloudwatch_receiver`.

   Linux

   ```
   <outputs formatid="fmtinfo">
      <console formatid="fmtinfo"/>
      <rollingfile type="size" filename="/var/log/amazon/ssm/amazon-ssm-agent.log"
    maxsize="30000000" maxrolls="5"/>
      <filter levels="error,critical" formatid="fmterror">
         <rollingfile type="size" filename="/var/log/amazon/ssm/errors.log"
    maxsize="10000000" maxrolls="5"/>
   ```

```
   </filter>
   <custom name="cloudwatch_receiver" formatid="fmtdebug" data-log-group="your-
CloudWatch-log-group-name"/>
</outputs>
```

Windows

```
<outputs formatid="fmtinfo">
   <console formatid="fmtinfo"/>
   <rollingfile type="size" maxrolls="5" maxsize="30000000"
 filename="{{LOCALAPPDATA}}\Amazon\SSM\Logs\amazon-ssm-agent.log"/>
   <filter formatid="fmterror" levels="error,critical">
      <rollingfile type="size" maxrolls="5" maxsize="10000000"
 filename="{{LOCALAPPDATA}}\Amazon\SSM\Logs\errors.log"/>
   </filter>
   <custom name="cloudwatch_receiver" formatid="fmtdebug" data-log-group="your-
CloudWatch-log-group-name"/>
</outputs>
```

5. Enregistrez vos modifications, puis redémarrez l'Agent SSM ou l'instance.
6. Ouvrez la console CloudWatch à l'adresse https://console.aws.amazon.com/cloudwatch/.
7. Dans le volet de navigation, choisissez Log groups (Groupes de journaux), puis choisissez le nom de votre groupe de journaux.

   Tip

   Le flux de journaux pour les données des fichiers Agent SSM journaux est organisé par ID d'instance.

Surveillance des métriques Fonctionnalité Exécuter la commande avec Amazon CloudWatch

Les métriques sont le concept fondamental de CloudWatch. Une métrique représente un ensemble de points de données chronologiques qui sont publiés dans CloudWatch. Envisagez une métrique comme une variable à surveiller et les points de données comme les valeurs de cette variable au fil du temps.

AWS Systems Manager publie des métriques sur le statut des commandes Fonctionnalité Exécuter la commande dans CloudWatch, ce qui vous permet de définir des alarmes en fonction de ces métriques. Ces statistiques sont enregistrées pendant une période prolongée afin que vous puissiez accéder à des informations historiques et avoir une meilleure idée du taux de réussite des commandes exécutées dans votre compte AWS.

Les valeurs de statut terminal des commandes pour lesquelles vous pouvez suivre les métriques incluent `Success` , et `Failed` .`Delivery Timed Out` Par exemple, pour un document de commande SSM défini pour s'exécuter toutes les heures, vous pouvez configurer une alarme pour vous avertir lorsqu'un statut de n'`Success`est pas signalé pour l'une de ces heures. Pour plus d'informations sur les valeurs de statut des commandes, consultez Comprendre les états des commandes (p. 965).

Pour afficher les métriques dans la console CloudWatch

1. Ouvrez la console CloudWatch à l'adresse https://console.aws.amazon.com/cloudwatch/.
2. Dans le volet de navigation, sélectionnez Metrics (Métriques).
3. Sélectionnez la vignette SSM Run Command.

Pour afficher les métriques à l'aide de l'AWS CLI

Ouvrez une invite de commande et utilisez la commande suivante :

```
aws cloudwatch list-metrics --namespace "AWS/SSM-RunCommand"
```

Pour répertorier toutes les métriques disponibles, utilisez la commande suivante :

```
aws cloudwatch list-metrics --namespace "AWS/SSM-RunCommand"
```

Dimensions et métriques Systems Manager Fonctionnalité Exécuter la commande

Systems Managerenvoie les métriques de Systems Manager Fonctionnalité Exécuter la commande commande à CloudWatch une seule fois toutes les minutes.

Systems Manager envoie les métriques de commande suivantes à CloudWatch.

> **Note**
>
> Toutes ces métriques utilisent `Count` comme unité, par conséquent, `Sum` et `SampleCount` sont les statistiques les plus utiles.

Métrique	Description
`CommandsDeliveryTimedOut`	Nombre de commandes dont le statut final est `Delivery Timed Out`.
`CommandsFailed`	Nombre de commandes dont le statut final est `Failed`.
`CommandsSucceeded`	Nombre de commandes dont le statut final est `Success`.

Pour plus d'informations sur l'utilisation des métriques CloudWatch, consultez les rubriques suivantes dans le Guide de l'utilisateur Amazon CloudWatch :

- Métriques
- Utilisation de métriques Amazon CloudWatch
- Utilisation d'Amazon CloudWatchalarmes

Journalisation des appels d'API AWS Systems Manager avec AWS CloudTrail

Systems Manager est intégré à AWS CloudTrail, un service qui fournit un enregistrement des actions effectuées par un utilisateur, un rôle ou un service AWS dans Systems Manager. CloudTrail capture tous les appels d'API pour Systems Manager en tant qu'événements, y compris les appels depuis la console Systems Manager et depuis les appels de code aux API Systems Manager. Si vous créez un journal de suivi, vous pouvez activer la livraison continue d'événements CloudTrail dans un compartiment S3, y compris d'événements pour Systems Manager. Si vous ne configurez pas de journal de suivi, vous pouvez toujours afficher les événements les plus récents dans la console CloudTrail dans Event history (Historique

des événements). À l'aide des informations collectées par CloudTrail, vous pouvez déterminer la demande qui a été envoyée à Systems Manager, l'adresse IP source à partir de laquelle la demande a été effectuée, l'auteur de la demande et la date de la demande, ainsi que d'autres informations.

Pour de plus amples informations sur CloudTrail, veuillez consulter AWS CloudTrail User Guide.

Informations Systems Manager dans CloudTrail

CloudTrail est activé sur votre compte AWS lorsque vous créez le compte. Lorsqu'une activité a lieu dans Systems Manager, cette activité est enregistrée dans un événement CloudTrail avec d'autres événements de services AWS dans Event history (Historique des événements). Vous pouvez afficher, rechercher et télécharger les événements récents dans votre compte AWS. Pour plus d'informations, consultez Affichage des événements avec l'historique des événements CloudTrail.

Pour un enregistrement continu des événements dans votre compte AWS, y compris les événements pour Systems Manager, créez un journal de suivi. Un journal de suivi permet à CloudTrail de livrer les fichiers journaux dans un compartiment S3. Par défaut, lorsque vous créez un journal de suivi dans la console, il s'applique à toutes les régions. Le journal de suivi consigne les événements de toutes les régions dans la partition AWS et livre les fichiers journaux dans le compartiment S3 de votre choix. En outre, vous pouvez configurer d'autres services AWS pour analyser plus en profondeur les données d'événements collectées dans les journaux CloudTrail et agir sur celles-ci. Pour plus d'informations, veuillez consulter:

- Présentation de la création d'un journal de suivi
- CloudTrail Services et intégrations pris en charge
- Configuration des notifications Amazon SNS pour CloudTrail
- Recevoir les fichiers journaux de CloudTrail de plusieurs régions et Recevoir les fichiers journaux de CloudTrail de plusieurs comptes

Toutes les actions Systems Manager sont consignées par CloudTrail et documentées dans l' Référence d'API AWS Systems Manager. Par exemple, les appels adressés aux actions `CreateMaintenanceWindows`, `PutInventory`, `SendCommand` et `StartSession` génèrent des entrées dans les fichiers journaux CloudTrail. (Pour voir un exemple de configuration de CloudTrail pour surveiller un appel d'API Systems Manager, consultez Surveillance de l'activité de la session avec Amazon EventBridge (console) (p. 933).)

Chaque événement ou entrée du journal contient des informations sur la personne qui a généré la demande. Les informations relatives à l'identité permettent de déterminer les éléments suivants :

- Si la demande a été effectuée avec les informations d'identification utilisateur racine ou IAM.
- Si la demande a été effectuée avec des informations d'identification de sécurité temporaires pour un rôle ou un utilisateur fédéré.
- Si la demande a été effectuée par un autre service AWS.

Pour plus d'informations, consultez la section Élément userIdentity CloudTrail.

Présentation des entrées des fichiers journaux Systems Manager

Un journal de suivi est une configuration qui active la livraison d'événements en tant que fichiers journaux à un compartiment S3 que vous spécifiez. Les fichiers journaux CloudTrail contiennent une ou plusieurs entrées de journal. Un événement représente une demande individuelle à partir d'une source quelconque et comprend des informations sur l'action demandée, la date et l'heure de l'action, les paramètres de la

demande, etc. Les fichiers journaux CloudTrail ne sont pas des séries ordonnées retraçant les appels d'API publics. Ils ne suivent donc aucun ordre précis.

L'exemple suivant présente une entrée de journal CloudTrail qui illustre l'action `DeleteDocuments` sur un document nommé `example-Document` dans la région Région USA Est (Ohio) (us-east-2).


```
{
    "eventVersion": "1.04",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AKIAI44QH8DHBEXAMPLE:203.0.113.11",
        "arn": "arn:aws:sts::123456789012:assumed-role/example-role/203.0.113.11",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2018-03-06T20:19:16Z"
            },
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AKIAI44QH8DHBEXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/example-role",
                "accountId": "123456789012",
                "userName": "example-role"
            }
        }
    },
    "eventTime": "2018-03-06T20:30:12Z",
    "eventSource": "ssm.amazonaws.com",
    "eventName": "DeleteDocument",
    "awsRegion": "us-east-2",
    "sourceIPAddress": "203.0.113.11",
    "userAgent": "example-user-agent-string",
    "requestParameters": {
        "name": "example-Document"
    },
    "responseElements": null,
    "requestID": "86168559-75e9-11e4-8cf8-75d18EXAMPLE",
    "eventID": "832b82d5-d474-44e8-a51d-093ccEXAMPLE",
    "resources": [
        {
            "ARN": "arn:aws:ssm:us-east-2:123456789012:document/example-Document",
            "accountId": "123456789012"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "123456789012"
}
```


Configuration d'Amazon CloudWatch Logs pour Fonctionnalité Exécuter la commande

Lorsque vous envoyez une commande à l'aide de la fonctionnalité Fonctionnalité Exécuter la commande, vous pouvez spécifier où vous souhaitez envoyer la sortie de la commande. Par défaut, Systems Manager renvoie uniquement les 2 500 premiers caractères de la sortie de commande. Si vous souhaitez afficher les détails de la sortie de la commande, vous pouvez spécifier un compartiment Amazon Simple Storage Service (Amazon S3). Ou vous pouvez spécifier Amazon CloudWatch Logs. Si vous spécifiez CloudWatch Logs, la fonctionnalité Fonctionnalité Exécuter la commande envoie périodiquement toutes les sorties de

commande et les journaux d'erreurs à CloudWatch Logs. Vous pouvez surveiller les journaux de sortie quasiment en temps réel, rechercher des phrases, valeurs ou modèles spécifiques, et créer des alarmes sur la base de la recherche.

Si vous avez configuré votre instance ou votre machine hybride sur site pour utiliser les stratégies gérées AWS Identity and Access Management (IAM) AmazonSSMManagedInstanceCore et CloudWatchAgentServerPolicy votre instance ne nécessite aucune configuration supplémentaire pour envoyer la sortie vers CloudWatch Logs . Vous avez simplement besoin de choisir cette option si vous envoyez des commandes à partir de la console, ou d'ajouter la section `cloud-watch-output-config` et le paramètre `CloudWatchOutputEnabled` si vous utilisez l'AWS CLI, les Outils pour Windows PowerShell ou une action d'API. La section `cloud-watch-output-config` et le paramètre `CloudWatchOutputEnabled` sont décrits plus en détail ultérieurement dans cette rubrique.

Pour de plus amples informations sur l'ajout de stratégies à un profil d'instance pour les instances EC2, veuillez consulter Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). Pour de plus amples informations sur l'ajout de stratégies à un rôle de service pour les instances et les machines virtuelles sur site que vous prévoyez d'utiliser en tant qu'instances gérées, veuillez consulter Créer un rôle de service IAM pour un environnement hybride (p. 50).

Pour de plus amples informations sur la mise à jour d'un profil d'instance existant, veuillez consulter Pour ajouter des autorisations au profil d'instance Systems Manager (console) (p. 38).

Si vous utilisez une stratégie personnalisée sur vos instances, vous devez mettre à jour la stratégie au niveau de chaque instance afin d'autoriser Systems Manager pour envoyer la sortie et les journaux vers les CloudWatch Logs. Ajoutez les objets de stratégie suivants à votre stratégie personnalisée. Pour de plus amples informations sur la mise à jour d'une stratégie IAM, veuillez consulter Modification de stratégies IAM dans le IAM Guide de l'utilisateur.

```
{
   "Effect":"Allow",
   "Action":[
      "logs:CreateLogGroup",
      "logs:CreateLogStream",
      "logs:DescribeLogGroups",
      "logs:DescribeLogStreams",
      "logs:PutLogEvents"
   ],
   "Resource":"arn:aws:logs:::log-group:/aws/ssm/*"
},
```

Spécification de CloudWatch Logs lorsque vous envoyez des commandes

Pour spécifier CloudWatch Logs en tant que sortie lorsque vous envoyez une commande à partir de l' AWS Management Console , choisissez CloudWatch Sortie dans la section Output options (Options de sortie). Si vous le souhaitez, vous pouvez spécifier le nom du groupe CloudWatch Logs dans lequel vous souhaitez envoyer la sortie de commande. Si vous ne spécifiez pas de nom de groupe, Systems Manager crée automatiquement un groupe de journaux pour vous. Le groupe de journaux utilise le format de dénomination suivant : /aws/ssm/*SystemsManagerDocumentName*.

Si vous exécutez des commandes à l'aide de l'AWS CLI, vous devez spécifier la section `cloud-watch-output-config` dans votre commande. Cette section vous permet de spécifier le paramètre `CloudWatchOutputEnabled`, et éventuellement le paramètre `CloudWatchLogGroupName`. Voici un exemple :

```
aws ssm send-command --document-name "AWS-RunPowerShellScript" --parameters commands=["echo
 helloWorld"] --targets "Key=instanceids,Values=an instance ID" --cloud-watch-output-config
 '{"CloudWatchLogGroupName":"log group name","CloudWatchOutputEnabled":true}'
```

Affichage de la sortie de commande dans CloudWatch Logs

Dès que la commande s'exécute, Systems Manager envoie la sortie dans CloudWatch Logs quasiment en temps réel. La sortie dans CloudWatch Logs est au format suivant :

CommandID/*InstanceID*/*PluginID*/stdout

CommandID/*InstanceID*/*PluginID*/stderr

Le résultat de l'exécution est chargé toutes les 30 secondes ou lorsque le tampon dépasse 200 Ko, selon la situation qui survient en premier.

> **Note**
>
> Les flux de journaux sont uniquement créés lorsque les données de sortie sont disponibles. Par exemple, s'il n'y a pas de données d'erreur pour une exécution, le flux stderr n'est pas créé.

Voici un exemple de la sortie de la commande tel qu'elle apparaît dans CloudWatch Logs.

```
Group - /aws/ssm/AWS-RunShellScript
Streams -
1234-567-8910/i-abcd-efg-hijk/AWS-RunPowerShellScript/stdout
24/1234-567-8910/i-abcd-efg-hijk/AWS-RunPowerShellScript/stderr
```

Surveillance des événements Systems Manager avec Amazon EventBridge

Amazon EventBridge est un service de bus d’événements sans serveur qui facilite la connexion de vos applications avec des données provenant de diverses sources. EventBridge fournit un flux de données en temps réel à partir de vos propres applications, applications SaaS (logiciel en tant que service) et AWS et achemine ces données vers des cibles telles que AWS Lambda. Vous pouvez configurer des règles de routage pour déterminer où envoyer vos données pour créer des architectures d'application qui réagissent en temps réel à toutes vos sources de données. EventBridge vous permet de créer des architectures pilotées par les événements, qui sont faiblement couplées et distribuées.

EventBridge était auparavant appelé Amazon CloudWatch Events. Il inclut de nouvelles fonctionnalités qui vous permettent de recevoir des événements de partenaires SaaS et de vos propres applications. Les utilisateurs CloudWatch Events existants peuvent accéder à leurs bus, règles et événements par défaut existants dans la nouvelle console EventBridge et dans la console CloudWatch Events. EventBridge utilise la même API CloudWatch Events, de sorte que toute votre utilisation existante de l'APICloudWatch Events reste la même.

En tout, EventBridge peut ajouter des événements de près de 40 AWS vers vos règles et cibles à partir de près de 20 services AWS.

EventBridge fournit une assistance pour les deux AWS Systems Manager des événements et Systems Manager cibles.

Pris en charge Systems Manager types d’événements

Parmi les nombreux types de Systems Manager les événements qui EventBridge peut détecter les éléments suivants :

- Une fenêtre de maintenance est désactivée
- Un workflow d’automatisation terminé avec succès

- Une instance gérée n'étant pas conforme aux correctifs
- Une valeur de paramètre en cours de mise à jour

EventBridge prend en charge les événements des suivants Systems Manager capacités :

- Inventaire
- Fenêtres de maintenance
- Parameter Store
- Fonctionnalité Exécuter la commande
- State Manager

Pour plus de détails sur les Systems Manager types d'événements, voir Référence : modèles et types Amazon EventBridge d'événements pourSystems Manager (p. 1387) et AWS Systems Manager Événements dans le Guide de l'utilisateur Amazon EventBridge.

Pris en charge Systems Manager types de cibles

EventBridge prend en charge les trois éléments suivants Systems Manager en tant que cibles d'une règle d'événement :

- Exécution d'un flux de travail d'automatisation
- Exécution d'un Fonctionnalité Exécuter la commande Document de commande
- Création d'un OpsCenter OpsItem

Pour connaître les méthodes suggérées pour utiliser ces cibles, consultez Systems Manager exemples de type de cible (p. 1343).

Pour plus d'informations sur la façon de démarrer avec EventBridge et configurer des règles, voir Mise en route avec Amazon EventBridge dans le Guide de l'utilisateur Amazon EventBridge. Pour plus d'informations sur l'utilisation de EventBridge, voir le Guide de l'utilisateur Amazon EventBridge.

Rubriques

- Configuration de EventBridge pour Systems Manager événements (p. 1341)
- Systems Manager exemples de type de cible (p. 1343)

Configuration de EventBridge pour Systems Manager événements

Vous pouvez utiliser Amazon EventBridge pour effectuer un événement cible lorsqu'il est pris en charge AWS Systems Manager des changements d'état, des changements d'état ou d'autres conditions se produisent. Vous pouvez créer une règle qui s'exécute chaque fois qu'il y a un état ou une transition d'état, ou lorsqu'il y a une transition vers un ou plusieurs états qui présentent un intérêt.

La procédure suivante décrit les étapes générales de création d'un EventBridge règle qui s'engage lorsqu'un événement spécifié est émis par Systems Manager. Pour une liste des procédures de ce guide de l'utilisateur qui traitent de scénarios spécifiques, voir Contenu associé à la fin de cette rubrique.

> **Note**
>
> Lorsqu'un service AWS de votre compte émet un événement, il accède toujours au bus d'événement par défaut de votre compte. Pour écrire une règle qui répond aux événements à partir de AWS dans votre compte, vous devez l'associer au bus d'événement par défaut. Vous pouvez créer une règle sur un bus d'événement personnalisé qui recherche des événements à

partir de AWS services, mais cette règle ne s'engage que lorsque vous recevez un tel événement d'un autre compte via la livraison d'événements entre comptes. Pour plus d'informations, consultez Envoi et réception d'événements entre des comptes AWS dans le Guide de l'utilisateur Amazon EventBridge.

Pour configurer EventBridge pour Systems Manager événements

1. Ouvrez la console Amazon EventBridge à l'adresse https://console.aws.amazon.com/events/.
2. Dans le volet de navigation, choisissez Règles, puis choisissez Créer une règle.

   -ou-

   Si le Amazon EventBridge page d'accueil s'ouvre en premier, choisissez Créer une règle.
3. Saisissez un nom et une description pour la règle.

   Une règle ne peut pas avoir le même nom qu'une autre règle de la même AWS Région et sur le même bus d'événement.
4. Pour Define pattern (Définir un modèle), choisissez Event pattern (Modèle d'événement).
5. Choisissez Pré-defined pattern by service (Modèle prédéfini par service).
6. Pour Service provider (Fournisseur de services), choisissez AWS.
7. Pour Nom du service, choisissez EC2 simple Systems Manager (SSM).
8. Pour Type d'événement, effectuez l'une des opérations suivantes :

   - Choisir Tous les événements.

     Si vous choisissez Tous les événements, tous les événements émis par Systems Manager correspondra à la règle. Notez que cette option peut entraîner un très grand nombre d'actions cibles d'événement.
   - Choisissez le type de Systems Manager événement à utiliser pour cette règle. EventBridge prend actuellement en charge les événements de Systems Manager capacités :
     - Conformité de la configuration
     - Inventaire
     - Fenêtres de maintenance
     - Parameter Store
     - Fonctionnalité Exécuter la commande
     - State Manager

       Note

       Pour Systems Manager actions qui ne sont pas prises en charge par EventBridge, vous pouvez choisir Appel d'API AWS via CloudTrail pour créer une règle d'événement basée sur un appel d'API, qui sont enregistrés par AWS CloudTrail. Pour obtenir un exemple, veuillez consulter Surveillance de l'activité de la session avec Amazon EventBridge (console) (p. 933).

   (Facultatif) Si vous souhaitez personnaliser le modèle d'événement, choisissez Modifier à côté de Modèle d'événement, effectuez vos modifications et choisissez Économiser.
9. Si vous avez choisi un Systems Manager à l'étape 8, effectuez l'une des opérations suivantes :

   - Pour les cibles à appeler lorsqu'un type d'événement pris en charge pour la capacité sélectionnée se produit, choisissez Tout type de détail.
   - Pour les cibles à appeler lorsque seuls certains types d'événements pour la capacité sélectionnée se produisent, choisissez Type(s) de détail spécifique(s), puis choisissez un ou plusieurs types dans la liste.

Pour plus de détails sur les types de détails pris en charge, voir Référence : modèles et types Amazon EventBridge d'événements pourSystems Manager (p. 1387) et AWS Systems Manager Événements dans le Guide de l'utilisateur Amazon EventBridge.

10. Si vous avez choisi un Systems Manager à l'étape 8, choisissez d'appeler des cibles pour certains types de détails, états ou d'autres options prises en charge. Les options disponibles dépendent de la capacité que vous avez sélectionnée.
11. Pour Select event bus (Sélectionner un bus d'événement), choisissez le bus d'événement que vous souhaitez associer à cette règle. Si vous voulez que cette règle réponde aux événements correspondants qui proviennent de votre propre AWS compte, sélectionnez Bus d'événement AWS par défaut. Lorsqu'un service AWS de votre compte émet un événement, il accède toujours au bus d'événement par défaut de votre compte.
12. Pour Select targets (Sélectionner les cibles),choisissez le service AWS qui doit agir lorsqu'un événement du type sélectionné est détecté.
13. Dans les autres champs de cette section, entrez des informations spécifiques à ce type de cible, le cas échéant.
14. Pour de nombreux types de cibles, EventBridge a besoin d'autorisations pour envoyer les événements à la cible. Dans ces cas, EventBridge peut créer le AWS Identity and Access Management (IAM) rôle nécessaire pour l'exécution de votre règle :
    - Pour créer un rôle IAM automatiquement, choisissez Create a new role for this specific resource.
    - Pour utiliser un IAM que vous avez créé précédemment, choisissez Utiliser le rôle existant.
15. Le cas échéant, choisissez Add target (Ajouter une cible) pour ajouter une autre cible pour cette règle.
16. (Facultatif) Entrez une ou plusieurs balises pour la règle. Pour de plus amples informations, veuillez consulter Balisage de vos ressources Amazon EventBridge dans le Guide de l'utilisateur Amazon EventBridge.
17. Sélectionnez Create.

Contenu connexe

- Création d'un EventBridge événement qui exécute une automatisation (console) (p. 344)
- Procédure : Utilisation des transformateurs en entrée avec Automation (p. 666)
- Correction des problèmes de conformité à l'aide d'EventBridge (p. 785)
- Affichage des actions de suppression d'inventaire dans EventBridge (p. 834)
- Configuration de EventBridge pour créer automatiquement des OpsItems pour les événements spécifiques (p. 188)
- Configuration de EventBridge pour les paramètres (p. 239)
- Configuration de EventBridge pour les stratégies de paramètres (p. 241)

Systems Manager exemples de type de cible

Lorsque vous spécifiez la cible à appeler dans un Amazon EventBridge , vous pouvez choisir parmi plus de 20 types de cibles et ajouter jusqu'à cinq cibles à chaque règle.

des différentes cibles, vous pouvez choisir parmi trois AWS Systems Manager en tant qu'actions cibles lorsqu'un EventBridge se produit lors de l'événement. Ces capacités incluent l'automatisation, OpsCenter, et Fonctionnalité Exécuter la commande.

Voici quelques exemples de façons d'utiliser ces capacités comme cible d'un EventBridge règle.

Exemples d'automatisation

Vous pouvez configurer un EventBridge pour démarrer les flux de travail Automation lorsque des événements tels que les suivants se produisent :

- Lorsqu'un Amazon CloudWatch signale qu'une instance gérée a échoué à une vérification d'état (`StatusCheckFailed_Instance=1`), exécutez le `AWSSupport-ExecuteEC2Rescue` Document d'automatisation sur l'instance.
- Lorsqu'un `EC2 Instance State-change Notification` se produit car un nouvel événement Amazon Elastic Compute Cloud (Amazon EC2) instance est en cours d'exécution, exécutez le `AWS-AttachEBSVolume` Document d'automatisation sur l'instance.
- Lorsqu'un Amazon Elastic Block Store (Amazon EBS) est créé et disponible, exécutez le `AWS-CreateSnapshot` Document d'automatisation sur le volume.

Exemples avec le kit OpsCenter

Vous pouvez configurer un EventBridge pour créer une nouvelle règle OpsItem lorsque des incidents tels que les suivants se produisent :

- Un événement de limitation pour Amazon DynamoDB se produit, ou Amazon EBS Les performances du volume ont dégradé.
- Un Amazon EC2 Auto Scaling ne parvient pas à lancer une instance ou un Systems Manager Échec du flux de travail d'automatisation.
- Une instance EC2 change d'état `Running` à `Stopped`.

Exemples avec le kit Fonctionnalité Exécuter la commande

Vous pouvez configurer un EventBridge pour exécuter une règle Systems Manager Document de commande dans Fonctionnalité Exécuter la commande lorsque des événements tels que les suivants se produisent :

- Lorsqu'un Auto Scaling est sur le point de se terminer, un Fonctionnalité Exécuter la commande peut capturer les fichiers journaux à partir de l'instance avant qu'elle ne soit résiliée.
- Lorsqu'une nouvelle instance est créée dans un Auto Scaling groupe, un Fonctionnalité Exécuter la commande cible peut activer le rôle de serveur Web ou installer le logiciel sur l'instance.
- Lorsqu'une instance gérée est jugée non conforme, un Fonctionnalité Exécuter la commande de l'action cible peut mettre à jour les correctifs sur l'instance en exécutant le `AWS-RunPatchBaseline` document.

Surveillance des changements de statut Systems Manager à l'aide de notifications Amazon SNS

Vous pouvez configurer Amazon Simple Notification Service (Amazon SNS) pour envoyer des notifications sur le statut des commandes que vous envoyez à l'aide de la fonctionnalité AWS Systems Manager Fonctionnalité Exécuter la commande ou d'une Fenêtres de maintenance AWS Systems Manager. Amazon SNS coordonne et gère l'envoi et la diffusion des notifications aux clients ou aux points de terminaison qui sont abonnés à des rubriques Amazon SNS. Vous pouvez recevoir une notification chaque fois qu'une commande change de statut ou passe à un statut spécifique, par exemple, Échec ou Expiré. Lorsque vous envoyez une commande à plusieurs instances, vous pouvez recevoir une notification pour chaque copie de la commande envoyée à une instance spécifique. Chaque copie s'appelle un appel.

Amazon SNS peut envoyer des notifications HTTP ou HTTPS POST, par e-mail (SMTP, texte brut ou format JSON) ou via un message publié dans une file d'attente Amazon Simple Queue Service (Amazon SQS). Pour plus d'informations, consultez Qu'est-ce que Amazon SNS dans le manuel Amazon Simple Notification Service Manuel du développeur. Pour voir des exemples de la structure des données JSON incluses dans la notification Amazon SNS fournie par la fonctionnalité Fonctionnalité Exécuter la

commande et une Fenêtres de maintenance, veuillez consulter Exemples de notifications Amazon SNS pour AWS Systems Manager (p. 1351).

Configurer les notifications Amazon SNS pour AWS Systems Manager

La fonctionnalité Fonctionnalité Exécuter la commande et les tâches Fonctionnalité Exécuter la commande qui sont enregistrées auprès d'une fenêtre de maintenance peuvent envoyer des notifications Amazon SNS pour les tâches de commande dont le statut devient l'un des suivants. Pour de plus amples informations sur les conditions qui entraînent ces changements de statut, veuillez consulter la section Comprendre les états des commandes (p. 965).

- En cours
- Success (Succès)
- Failed (Échec)
- Expiré
- Canceled (Annulé)

> **Note**
>
> Les commandes envoyées à l'aide de la fonctionnalité Fonctionnalité Exécuter la commande signalent également les statuts Annulation en cours et En suspens. Ces statuts ne sont pas enregistrés par les notifications Amazon SNS.

Récapitulatif des commandes pour les notifications Amazon SNS

Si vous configurez la fonctionnalité Fonctionnalité Exécuter la commande ou une tâche Fonctionnalité Exécuter la commande dans votre fenêtre de maintenance pour les notifications Amazon SNS, Amazon SNS envoie des messages récapitulatifs qui incluent les informations suivantes.

Field	Tapez	Description
eventTime	String (chaîne)	Heure à laquelle l'événement a été déclenché. L'horodatage est important, car Amazon SNS ne garantit pas l'ordre de transmission des messages. Exemple: 2016-04-26T13:15:30 Z
documentName	String (chaîne)	Nom du document SSM utilisé pour exécuter cette commande.
commandId	String (chaîne)	ID généré par la fonctionnalité Fonctionnalité Exécuter la commande une fois que la commande a été envoyée.
expiresAfter	Date	Si la date d'expiration est atteinte et que la commande n'a pas encore été exécutée, l'exécution n'aura pas lieu.
outputS3BucketName	String (chaîne)	Compartiment Amazon Simple Storage Service (Amazon S3)

Field	Tapez	Description
		dans lequel les réponses à l'exécution de la commande doivent être stockées.
outputS3KeyPrefix	String (chaîne)	Chemin du répertoire Amazon S3 à l'intérieur du compartiment dans lequel les réponses à l'exécution de la commande doivent être stockées.
requestedDateTime	String (chaîne)	Heure et date auxquelles la demande a été envoyée à cette instance spécifique.
instanceIds	StringList	Instances qui étaient ciblées par la commande. Note IDsLes instances ne sont incluses dans le message récapitulatif que si l'instance ciblée par la Fonctionnalité Exécuter la commande tâche IDs est directement incluse. Les instances ne IDs sont pas incluses dans le message récapitulatif si la Fonctionnalité Exécuter la commande tâche a été émise à l'aide d'un ciblage basé sur des balises.
status	String (chaîne)	Statut de la commande.

Notifications Amazon SNS basées sur les appels

Si vous envoyez une commande à plusieurs instances, Amazon SNS peut envoyer des messages concernant chaque copie ou appel de la commande. Les messages incluent les informations suivantes.

Field	Tapez	Description
eventTime	String (chaîne)	Heure à laquelle l'événement a été déclenché. L'horodatage est important, car Amazon SNS ne garantit pas l'ordre de transmission des messages. Exemple: 2016-04-26T13:15:30 Z
documentName	String (chaîne)	Nom du document Systems Manager utilisé pour exécuter cette commande.

Field	Tapez	Description
requestedDateTime	String (chaîne)	Heure et date auxquelles la demande a été envoyée à cette instance spécifique.
commandId	String (chaîne)	ID généré par la fonctionnalité Fonctionnalité Exécuter la commande une fois que la commande a été envoyée.
instanceId	String (chaîne)	Instance qui était ciblée par la commande.
status	String (chaîne)	Statut de la commande pour cet appel.

Pour configurer les notifications Amazon SNS en cas de changement de statut d'une commande, vous devez effectuer les tâches suivantes.

> **Note**
>
> Si vous ne configurez pas les notifications Amazon SNS pour votre fenêtre de maintenance, vous pouvez ignorer la tâche 5 ci-dessous.

Rubriques

- Tâche 1 : Créer une rubrique Amazon SNS et s'y abonner (p. 1347)
- Tâche 2 : Créer une IAM stratégie pour les Amazon SNS notifications (p. 1348)
- Tâche 3 : Créer un IAM rôle pour les Amazon SNS notifications (p. 1348)
- Tâche 4 : Configurer l'accès utilisateur (p. 1349)
- Tâche 5 : Attachement de la stratégie iam:PassRole à votre rôle de fenêtre de maintenance (p. 1350)

Tâche 1 : Créer une rubrique Amazon SNS et s'y abonner

Une rubrique Amazon SNS est un canal de communication que la fonctionnalité Fonctionnalité Exécuter la commande et les tâches Fonctionnalité Exécuter la commande qui sont enregistrées auprès d'une fenêtre de maintenance utilisent pour envoyer des notifications sur le statut de vos commandes. Amazon SNS prend en charge différents protocoles de communication, notamment HTTP/S, les e-mails et d'autres services AWS tels que Amazon SQS. Pour démarrer rapidement, nous vous recommandons de commencer par le protocole de messagerie. Pour de plus amples informations sur la création d'une rubrique, veuillez consulter Créer une rubrique dans le Amazon Simple Notification Service Manuel du développeur.

> **Note**
>
> Une fois que vous avez créé la rubrique, copier l'ARN de la rubrique ou prenez-en note. Vous devez spécifier cet ARN lorsque vous envoyez une commande qui est configurée pour renvoyer des notifications de statut.

Une fois que vous avez créé la rubrique, vous devez vous y abonner en spécifiant un Point de terminaison. Si vous avez choisi le protocole de messagerie, le point de terminaison est l'adresse e-mail à laquelle vous souhaitez recevoir des notifications. Pour de plus amples informations sur la façon de s'abonner à une rubrique, veuillez consulter Abonnement d'un point de terminaison à une Amazon SNS rubrique dans le Amazon Simple Notification Service Manuel du développeur.

Amazon SNS envoie un e-mail de confirmation à partir d'AWS Notifications (Notifications AWS) à l'adresse e-mail que vous spécifiez. Ouvrez cet e-mail et choisissez le lien Confirm subscription (Confirmer l'abonnement).

AWS vous renverra un accusé de réception. Amazon SNS est maintenant configuré pour recevoir des notifications et envoyer un e-mail de notification à l'adresse spécifiée.

Tâche 2 : Créer une IAM stratégie pour les Amazon SNS notifications

Utilisez la procédure suivante afin de créer une stratégie AWS Identity and Access Management (IAM) personnalisée qui fournit les autorisations nécessaires pour déclencher des notifications Amazon SNS.

Pour créer une stratégie IAM personnalisée pour les notifications Amazon SNS

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation, choisissez Stratégies, puis Créer une stratégie. (Si un bouton Mise en route est affiché, choisissez-le, puis choisissez Créer une stratégie.)
3. Choisissez l'onglet JSON.
4. Remplacez le contenu par défaut par ce qui suit :

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "sns:Publish"
            ],
            "Resource": "sns-topic-arn"
        }
    ]
}
```

   sns-topic-arn représente l'ARN de la rubrique Amazon Simple Notification Service (Amazon SNS) existante à utiliser pour envoyer Amazon Simple Notification Service des notifications, au format par exemple :`arn:aws:sns::account-id:sns-topic-name`. `arn:aws:sns::123456789012:my-sns-topic`.
5. Choisissez Examiner une stratégie.
6. Sur la page Examiner une stratégie, dans le champ Nom, saisissez un nom pour la stratégie en ligne. Exemple : **`my-sns-publish-permissions`**.
7. (Facultatif) Dans le champ Description, saisissez une description pour la stratégie.
8. Choisissez Create policy (Créer une stratégie).

Tâche 3 : Créer un IAM rôle pour les Amazon SNS notifications

Utilisez la procédure suivante afin de créer un rôle IAM pour les notifications Amazon SNS. Ce rôle de service est utilisé par Systems Manager pour déclencher les notifications Amazon SNS.

Pour créer un rôle de service IAM pour les notifications Amazon SNS

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le volet de navigation, choisissez Rôles, puis Créer un rôle.

3. Sous Select type of trusted entity (Sélectionner le type d'entité de confiance), choisissez Service AWS.
4. Dans la section Choose a use case (Choisir un cas d'utilisation), choisissez Systems Manager, puis Next (Suivant) : Autorisations.
5. Sur la page Attacher des stratégies d'autorisations, cochez la case située à gauche du nom de la stratégie personnalisée que vous avez créée à la tâche 2. Exemple : **`my-sns-publish-permissions`**.
6. Choisissez Next (Suivant) : Balises.
7. (Facultatif) Ajoutez une ou plusieurs paires clé-valeur de balise pour organiser, suivre ou contrôler l'accès pour ce rôle, puis choisissez Suivant : Passez en revue.
8. Sur la page Révision, tapez un nom dans la zone Nom du rôle tel que **`my-sns-role`**.
9. (Facultatif) Modifiez la description du rôle par défaut pour refléter l'objectif de ce rôle. Par exemple: **`Triggers SNS topics on your behalf.`**
10. Sélectionnez Créer un rôle. Le système vous renvoie à la page Rôles.
11. Copiez le rôle ARN ou prenez-en note. Cet ARN de rôle est utilisé lorsque vous envoyez une commande qui est configurée pour renvoyer des notifications Amazon SNS.
12. Laissez la page Récapitulatif ouverte.

Tâche 4 : Configurer l'accès utilisateur

Si votre compte utilisateur, votre groupe ou votre rôle AWS Identity and Access Management (IAM) dispose des autorisations d'administrateur, vous avez accès à Systems Manager Fonctionnalité Exécuter la commande et à Fenêtres de maintenance. Si vous ne disposez pas des autorisations d'administrateur, un administrateur doit vous les donner en affectant la stratégie gérée `AmazonSSMFullAccess` ou une stratégie dotée d'autorisations de même type à votre compte, groupe ou rôle IAM.

Utilisez la procédure suivante afin de configurer un compte utilisateur pour utiliser Fonctionnalité Exécuter la commande et Fenêtres de maintenance. Si vous devez créer un compte utilisateur, veuillez consulter la section Création d'un utilisateur IAM dans votre compte AWS dans le IAM Guide de l'utilisateur.

Pour configurer l'accès utilisateur et associer la stratégie iam:PassRole à un compte utilisateur

1. Dans le panneau de navigation IAM, choisissez Utilisateurs, puis choisissez le compte utilisateur à configurer.
2. Sous l'onglet Autorisations, dans la liste des stratégies, vérifiez que la stratégie **`AmazonSSMFullAccess`** est répertoriée ou qu'une stratégie comparable autorise le compte à accéder à Systems Manager.
3. Choisissez Add inline policy (Ajouter une stratégie en ligne).
4. Dans la page Créer une stratégie, choisissez l'onglet Éditeur visuel.
5. Choisissez Service, puis choisissez IAM.
6. Choisissez Sélectionner des actions.
7. Dans la zone de texte Filtrer les actions, tapez **`PassRole`** , puis choisissez l'PassRoleoption.
8. Choisissez Ressources. Vérifiez que Spécifique est sélectionné, puis choisissez Add ARN (Ajouter l'ARN).
9. Dans le champ Specify ARN for role (Spécifier l'ARN du rôle), collez l'ARN du rôle Amazon SNS que vous avez copié à la fin de la tâche 3. Le système remplit automatiquement les champs Account (Compte) et Role name with path (Nom du rôle avec chemin d'accès).
10. Choisissez Add (Ajouter).
11. Choisissez Examiner une stratégie.
12. Sur la page Review Policy (Examiner une stratégie), tapez un nom, puis choisissez Create Policy (Créer une stratégie).

Tâche 5 : Attachement de la stratégie iam:PassRole à votre rôle de fenêtre de maintenance

Lorsque vous enregistrez une tâche Fonctionnalité Exécuter la commande auprès d'une fenêtre de maintenance, vous spécifiez un Amazon Resource Name (ARN) de rôle de service. Ce rôle de service est utilisé par Systems Manager pour exécuter des tâches enregistrées auprès de la fenêtre de maintenance. Pour configurer les notifications Amazon SNS pour une tâche Fonctionnalité Exécuter la commande enregistrée, vous devez attacher une stratégie `iam:PassRole` au rôle de service de fenêtre de maintenance spécifié. Si vous n'avez pas l'intention de configurer la tâche enregistrée pour les notifications Amazon SNS, cette tâche peut être ignorée.

La stratégie `iam:PassRole` autorise le rôle de service Fenêtres de maintenance à transmettre le rôle SNS créé au cours de la tâche 3 au service Amazon SNS. La procédure suivante montre comment attacher la stratégie `iam:PassRole` au rôle de service Fenêtres de maintenance.

> **Note**
>
> Vous devez utiliser un rôle de service personnalisé pour votre fenêtre de maintenance afin d'envoyer des notifications concernant les tâches Exécuter la commande enregistrées. Pour plus d'informations, consultez la section Dois-je utiliser un rôle lié à un service ou un rôle de service personnalisé pour exécuter des tâches de fenêtre de maintenance ? (p. 691).
> Si vous avez besoin de créer un rôle de service personnalisé, veuillez consulter l'une des rubriques suivantes :
>
> - Contrôle de l'accès aux fenêtres de maintenance (console) (p. 692)
> - Contrôler l'accès aux fenêtres de maintenance (AWS CLI) (p. 697)
> - Contrôler l'accès aux fenêtres de maintenance (Outils pour Windows PowerShell) (p. 703)

Pour attacher la stratégie iam:PassRole à votre rôle Fenêtres de maintenance

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation, choisissez Rôles, puis sélectionnez le rôle Amazon SNS créé au cours de la tâche 3.
3. Copiez l'ARN de rôle ou notez-le et revenez à la section Rôles de la console IAM.
4. Sélectionnez le rôle de service Fenêtres de maintenance personnalisé que vous avez créé (sous le nom de Rôle).
5. Sous Autorisations, vérifiez que la stratégie `AmazonSSMMaintenanceWindowRole` est répertoriée ou qu'une stratégie comparable accorde à la fenêtre de maintenance l'autorisation d'accéder à l'API Systems Manager.
6. Choisissez Add inline policy (Ajouter une stratégie en ligne).
7. Choisissez l'onglet Visual Editor.
8. Pour Service, choisissez IAM.
9. Pour Actions, choisissez PassRole .
10. Pour Ressources, choisissez Spécifique, puis Ajouter un ARN.
11. Dans la zone Spécifier l'ARN pour le rôle collez l'ARN du rôle Amazon SNS IAM créé dans la tâche 3, puis choisissez Ajouter.
12. Choisissez Examiner une stratégie.
13. Sur la page Vérifier la stratégie, spécifiez un nom pour la stratégie, puis choisissez PassRole Créer une stratégie.

Exemples de notifications Amazon SNS pour AWS Systems Manager

Vous pouvez configurer Amazon Simple Notification Service (Amazon SNS) pour envoyer des notifications sur le statut des commandes que vous envoyez à l'aide de la fonctionnalité AWS Systems Manager Fonctionnalité Exécuter la commande ou d'une Fenêtres de maintenance AWS Systems Manager.

> Note
>
> Ce guide ne traite pas de la configuration des notifications pour la fonctionnalité Fonctionnalité Exécuter la commande ou une Fenêtres de maintenance. Pour de plus amples informations sur la configuration de la fonctionnalité Fonctionnalité Exécuter la commande ou d'une Fenêtres de maintenance pour envoyer des notifications Amazon SNS sur le statut des commandes, veuillez consulter Configurer les notifications Amazon SNS pour AWS Systems Manager (p. 1345).

Les exemples suivants montrent la structure de la sortie JSON renvoyée par les notifications Amazon SNS configurées pour la fonctionnalité Fonctionnalité Exécuter la commande ou une Fenêtres de maintenance.

Exemple de sortie JSON pour les messages du récapitulatif des commandes en utilisant le ciblage par ID d'instance

```
{
    "commandId": "a8c7e76f-15f1-4c33-9052-0123456789ab",
    "documentName": "AWS-RunPowerShellScript",
    "instanceIds": [
        "i-1234567890abcdef0",
        "i-9876543210abcdef0"
    ],
    "requestedDateTime": "2019-04-25T17:57:09.17Z",
    "expiresAfter": "2019-04-25T19:07:09.17Z",
    "outputS3BucketName": "DOC-EXAMPLE-BUCKET",
    "outputS3KeyPrefix": "runcommand",
    "status": "InProgress",
    "eventTime": "2019-04-25T17:57:09.236Z"
}
```

Exemple de sortie JSON pour les messages du récapitulatif des commandes en utilisant le ciblage basé sur des balises

```
{
    "commandId": "9e92c686-ddc7-4827-b040-0123456789ab",
    "documentName": "AWS-RunPowerShellScript",
    "instanceIds": [],
    "requestedDateTime": "2019-04-25T18:01:03.888Z",
    "expiresAfter": "2019-04-25T19:11:03.888Z",
    "outputS3BucketName": "",
    "outputS3KeyPrefix": "",
    "status": "InProgress",
    "eventTime": "2019-04-25T18:01:05.825Z"
}
```

Exemple de sortie JSON pour les messages d'appel

```
{
    "commandId": "ceb96b84-16aa-4540-91e3-925a9a278b8c",
    "documentName": "AWS-RunPowerShellScript",
    "instanceId": "i-1234567890abcdef0",
    "requestedDateTime": "2019-04-25T18:06:05.032Z",
```

```
    "status": "InProgress",
    "eventTime": "2019-04-25T18:06:05.099Z"
}
```

Utiliser la fonctionnalité Fonctionnalité Exécuter la commande pour envoyer une commande qui renvoie des notifications de statut

Les procédures suivantes montrent comment utiliser l'AWS Command Line Interface (AWS CLI) ou la console AWS Systems Manager pour envoyer une tâche Fonctionnalité Exécuter la commande configurée pour renvoyer des notifications de statut.

Envoi d'une tâche Fonctionnalité Exécuter la commande qui renvoie des notifications (console)

Utilisez la procédure suivante pour envoyer via la fonctionnalité Fonctionnalité Exécuter la commande une commande qui est configurée pour renvoyer des notifications de statut à l'aide de la console Systems Manager.

Pour envoyer une commande qui renvoie des notifications (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fonctionnalité Exécuter la commande.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Fonctionnalité Exécuter la commande.
3. Choisissez Exécuter une commande.
4. Dans la liste Command document (Document de commande), choisissez un document Systems Manager.
5. Dans la section Command parameters (Paramètres de la commande), indiquez des valeurs pour les paramètres requis.
6. Dans la section Cibles, identifiez les instances sur lesquelles vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant manuellement les instances ou en indiquant un groupe de ressources.

   > Note
   >
   > Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.
7. Pour Autres paramètres :
   - Pour Comment (Commentaire), saisissez des informations à propos de cette commande.
   - Pour Délai (secondes), précisez le nombre de secondes durant lesquelles le système doit attendre avant de mettre en échec l'exécution de la commande globale.
8. (Facultatif) Pour Contrôle du débit :
   - Pour Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage d'instances sur lesquelles exécuter simultanément la commande.

Note

Si vous avez sélectionné des cibles en indiquant des balises appliquées à des instances gérées ou en indiquant des groupes de ressources AWS, et que vous n'êtes pas sûr du nombre d'instances ciblées, limitez le nombre d'instances pouvant exécuter simultanément le document en spécifiant un pourcentage.

- Pour Seuil d'erreur, spécifiez quand arrêter l'exécution de la commande sur d'autres instances après son échec sur un nombre ou un pourcentage d'instances. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les instances sur lesquelles la commande est en cours de traitement peuvent également envoyer des erreurs.

9. (Facultatif) Pour Output options (Options de sortie), pour enregistrer la sortie de la commande dans un fichier, cochez la case Write command output to an S3 bucket (Écrire la sortie de commande vers un compartiment S3). Entrez les noms de compartiment et de préfixe (dossier) dans les zones.

   Note

   Les autorisations S3 qui accordent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour de plus amples informations, veuillez consulter Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). En outre, si le compartiment S3 spécifié se trouve dans un compte AWS différent, assurez-vous que le profil d'instance associé à l'instance dispose des autorisations nécessaires pour écrire dans ce compartiment.

10. Dans la section SNS Notifications (Notifications SNS), sélectionnez Enable SNS notifications (Activer les notifications SNS).
11. Dans le champ Rôle IAM, tapez ou collez l'ARN du rôle IAM SNS que vous avez créé dans la tâche 3 de la rubrique Surveillance des changements de statut Systems Manager à l'aide de notifications Amazon SNS (p. 1344).
12. Dans le champ SNS topic (Rubrique SNS), saisissez ou collez l'ARN de la rubrique Amazon SNS à utiliser.
13. Dans le champ Notify me on (M'avertir de), sélectionnez les événements pour lesquels vous souhaitez recevoir des notifications.
14. Dans le champ Notify me for (M'avertir pour), définissez si vous souhaitez recevoir des notifications pour chaque copie d'une commande envoyée à plusieurs instances (appels) ou uniquement le récapitulatif des commandes.
15. Cliquez sur Exécuter.
16. Consultez votre messagerie pour voir s'il y a un message de Amazon SNS et ouvrez l'e-mail. Amazon SNS peut prendre quelques minutes pour envoyer le message.

Envoi d'une tâche Fonctionnalité Exécuter la commande qui renvoie des notifications (interface de ligne de commande)

Utilisez la procédure suivante pour envoyer via la fonctionnalité Fonctionnalité Exécuter la commande une commande qui est configurée afin de renvoyer des notifications de statut à l'aide de l'AWS CLI.

Pour envoyer une commande qui renvoie des notifications (interface de ligne de commande)

1. Ouvrez AWS CLI.
2. Spécifiez les paramètres dans la commande suivante pour cibler en fonction de l'instance gérée IDs :

```
aws ssm send-command --instance-ids "ID-1, ID-2" --document-name "Name"
 --parameters '{"commands":["input"]}' --service-role "SNSRoleARN" --
```

```
notification-config '{"NotificationArn":"SNSTopicName","NotificationEvents":
["All"],"NotificationType":"Command"}'
```

Exemple :

```
aws ssm send-command --instance-ids "i-02573cafcfEXAMPLE, i-0471e04240EXAMPLE"
 --document-name "AWS-RunPowerShellScript" --parameters '{"commands":
["Get-Process"]}' --service-role "arn:aws:iam::111122223333:role/
SNS_Role" --notification-config '{"NotificationArn":"arn:aws:sns:us-
east-1:111122223333:SNSTopic","NotificationEvents":
["All"],"NotificationType":"Command"}'
```

Autres commandes

Spécifiez les paramètres de la commande suivante pour cibler les instances gérées en utilisant des balises :

```
aws ssm send-command --targets "Key=tag:TagName,Values=TagKey" --document-name
 "Name" --parameters '{"commands":["input"]}' --service-role "SNSRoleARN" --
notification-config '{"NotificationArn":"SNSTopicName","NotificationEvents":
["All"],"NotificationType":"Command"}'
```

Exemple :

```
aws ssm send-command --targets "Key=tag:Environment,Values=Dev" --
document-name "AWS-RunPowerShellScript" --parameters '{"commands":
["Get-Process"]}' --service-role "arn:aws:iam::111122223333:role/
SNS_Role" --notification-config '{"NotificationArn":"arn:aws:sns:us-
east-1:111122223333:SNSTopic","NotificationEvents":
["All"],"NotificationType":"Command"}'
```

3. Appuyez sur Enter.
4. Consultez votre messagerie pour voir s'il y a un message de Amazon SNS et ouvrez l'e-mail. Amazon SNS peut prendre quelques minutes pour envoyer le message.

Pour de plus amples informations sur la configuration de la fonctionnalité Fonctionnalité Exécuter la commande à partir de la ligne de commande, veuillez consulter Amazon EC2 Systems Manager API Reference et la Référence relative à l'AWS CLI Systems Manager.

Utilisation d'une fenêtre de maintenance pour envoyer une commande qui renvoie des notifications de statut

Les procédures suivantes montrent comment enregistrer une tâche AWS Systems Manager Fonctionnalité Exécuter la commande auprès de votre fenêtre de maintenance à l'aide de la console Systems Manager ou de l'AWS Command Line Interface (AWS CLI). Ces procédures expliquent aussi comment configurer la tâche Fonctionnalité Exécuter la commande pour renvoyer des notifications de statut.

Avant de commencer

Si vous n'avez pas encore créé de fenêtre de maintenance ou de cibles enregistrées, veuillez consulter Utilisation des fenêtres de maintenance (console) (p. 707) pour connaître les étapes permettant de créer une fenêtre de maintenance et d'enregistrer des cibles.

Pour recevoir des notifications du service Amazon SNS, vous devez attacher une stratégie `iam:PassRole` au rôle de service Fenêtres de maintenance spécifié dans la tâche enregistrée. Si vous n'avez pas ajouté

d'autorisations `iam:PassRole` à votre rôle de service Fenêtres de maintenance, veuillez consulter Tâche 5 : Attachement de la stratégie iam:PassRole à votre rôle de fenêtre de maintenance (p. 1350).

Enregistrement d'une tâche Fonctionnalité Exécuter la commande auprès d'une fenêtre de maintenance qui renvoie des notifications (console)

Utilisez la procédure suivante pour enregistrer une tâche Fonctionnalité Exécuter la commande configurée afin de renvoyer des notifications de statut à votre fenêtre de maintenance à l'aide de la console Systems Manager.

Pour enregistrer une tâche Fonctionnalité Exécuter la commande auprès de votre fenêtre de maintenance qui renvoie des notifications (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fenêtres de maintenance.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Fenêtres de maintenance.
3. Sélectionnez la fenêtre de maintenance pour laquelle vous souhaitez enregistrer une tâche Fonctionnalité Exécuter la commande configurée pour envoyer des notifications Amazon SNS.
4. Choisissez Actions, puis Register Fonctionnalité Exécuter la commande task (Enregistrer une tâche).
5. Dans le champ Name (Nom), entrez un nom pour la tâche.
6. Dans le champ Description, entrez une description.
7. Dans la liste Document, choisissez un document de commande.
8. Dans la liste Task priority (Priorité de tâche), spécifiez une priorité pour cette tâche. 1 est la priorité la plus élevée. Les tâches d'une fenêtre de maintenance sont planifiées par ordre de priorité. Les tâches qui ont la même priorité sont planifiées en parallèle.
9. Dans la section Cibles, identifiez les instances sur lesquelles vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant manuellement les instances ou en indiquant un groupe de ressources.

   > **Note**
   >
   > Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.
10. (Facultatif) Pour Contrôle du débit :

    - Pour Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage d'instances sur lesquelles exécuter simultanément la commande.

      > **Note**
      >
      > Si vous avez sélectionné des cibles en indiquant des balises appliquées à des instances gérées ou en indiquant des groupes de ressources AWS, et que vous n'êtes pas sûr du nombre d'instances ciblées, limitez le nombre d'instances pouvant exécuter simultanément le document en spécifiant un pourcentage.
    - Pour Seuil d'erreur, spécifiez quand arrêter l'exécution de la commande sur d'autres instances après son échec sur un nombre ou un pourcentage d'instances. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les

instances sur lesquelles la commande est en cours de traitement peuvent également envoyer des erreurs.

11. Dans la zone IAM service role (Rôle de service IAM), choisissez le rôle de service Fenêtres de maintenance qui dispose des autorisations `iam:PassRole` sur le rôle SNS.

    Note

    Vous devez ajouter des autorisations `iam:PassRole` au rôle de Fenêtres de maintenance pour permettre à Systems Manager de transmettre le rôle SNS à Amazon SNS. Si vous n'avez pas ajouté d'autorisations `iam:PassRole`, veuillez consulter Tâche 5 dans la rubrique Surveillance des changements de statut Systems Manager à l'aide de notifications Amazon SNS (p. 1344).

12. (Facultatif) Pour Output options (Options de sortie), pour enregistrer la sortie de la commande dans un fichier, cochez la case Write command output to an S3 bucket (Écrire la sortie de commande vers un compartiment S3). Entrez les noms de compartiment et de préfixe (dossier) dans les zones.

    Note

    Les autorisations S3 qui accordent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour de plus amples informations, veuillez consulter Créer un rôle de profil d'instance IAM pour Systems Manager (p. 34). En outre, si le compartiment S3 spécifié se trouve dans un compte AWS différent, assurez-vous que le profil d'instance associé à l'instance dispose des autorisations nécessaires pour écrire dans ce compartiment.

13. Dans la section SNS notifications (Notifications SNS), sélectionnez Enable SNS Notifications (Activer les notifications SNS).
14. Dans la section IAM role (Rôle IAM), choisissez le rôle SNS devant déclencher les notifications Amazon SNS.
15. Dans la section SNS topic (Rubrique SNS), saisissez ou collez l'ARN de la rubrique Amazon SNS à utiliser.
16. Dans la section Event type (Type d'événement), sélectionnez les événements pour lesquels vous souhaitez recevoir des notifications.
17. Dans la section Notification type (Type de notification), choisissez de recevoir des notifications pour chaque copie d'une commande envoyée à plusieurs instances (appels) ou le récapitulatif des commandes.
18. Dans la section Input Parameters (Paramètres d'entrée), entrez les paramètres requis en fonction du document de commande que vous avez choisi.
19. Choisissez Register Fonctionnalité Exécuter la commande task (Enregistrer une tâche d'exécution de commande).
20. Après l'exécution suivante de votre fenêtre de maintenance, consultez votre messagerie afin de vérifier s'il y a un message de Amazon SNS et ouvrez l'e-mail. Amazon SNS peut mettre quelques minutes à envoyer le message.

Enregistrement d'une tâche Fonctionnalité Exécuter la commande auprès d'une fenêtre de maintenance qui renvoie des notifications (interface de ligne de commande)

Utilisez la procédure suivante pour enregistrer une tâche Fonctionnalité Exécuter la commande qui est configurée pour renvoyer des notifications de statut à votre fenêtre de maintenance à l'aide de l'AWS CLI.

Pour enregistrer une tâche Fonctionnalité Exécuter la commande auprès de votre fenêtre de maintenance qui renvoie des notifications (interface de ligne de commande)

Note

Afin de mieux gérer vos options de tâche, cette procédure utilise l'option de commande `--cli-input-json`, avec des valeurs d'option enregistrées dans un fichier JSON.

1. Sur votre ordinateur local, créez un fichier nommé `RunCommandTask.json`.
2. Collez le contenu suivant dans le fichier :

```
{
    "Name": "Name",
    "Description": "Description",
    "WindowId": "mw-0c50858d01EXAMPLE",
    "ServiceRoleArn": "arn:aws:iam::111122223333:role/MaintenanceWindowIAMRole",
    "MaxConcurrency": "1",
    "MaxErrors": "1",
    "Priority": 3,
    "Targets": [
        {
            "Key": "WindowTargetIds",
            "Values": [
                "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
            ]
        }
    ],
    "TaskType": "RUN_COMMAND",
    "TaskArn": "CommandDocumentName",
    "TaskInvocationParameters": {
        "RunCommand": {
            "Comment": "Comment",
            "TimeoutSeconds": 3600,
            "NotificationConfig": {
                "NotificationArn": "arn:aws:sns:region:123456789012:SNSTopicName",
                "NotificationEvents": [
                    "All"
                ],
                "NotificationType": "Command"
            },
            "ServiceRoleArn": "arn:aws:iam::123456789012:role/SNSIAMRole"
        }
    }
}
```

3. Remplacez les exemples de valeurs par les informations sur vos propres ressources.

   Vous pouvez également restaurer les options que nous n'avons pas spécifiées dans cet exemple si vous souhaitez les utiliser. Par exemple, vous pouvez enregistrer la sortie de la commande dans un compartiment S3.

   Pour de plus amples informations, veuillez consulter register-task-with-maintenance-window.
4. Enregistrez le fichier .
5. Dans le répertoire de votre ordinateur local où vous avez enregistré le fichier, exécutez la commande suivante :

   ```
   aws ssm register-task-with-maintenance-window --cli-input-json file://
   RunCommandTask.json
   ```

> **Important**
>
> N'oubliez pas d'inclure `file://` devant le nom du fichier. Il est nécessaire dans cette commande.

Si elle aboutit, la commande renvoie des informations similaires à ce qui suit :

```
{
    "WindowTaskId": "j2l8d5b5c-mw66-tk4d-r3g9-1d4d1EXAMPLE"
}
```

6. Après l'exécution suivante de votre fenêtre de maintenance, consultez votre messagerie afin de vérifier s'il y a un message de Amazon SNS et ouvrez l'e-mail. Amazon SNS peut mettre quelques minutes à envoyer le message.

Pour de plus amples informations sur l'enregistrement de tâches pour la fenêtre de maintenance à partir de la ligne de commande, veuillez consulter Amazon EC2 Systems Manager API Reference et la Systems ManagerRéférence relative à l'interface de ligne de commande AWS.

Balisage des ressources Systems Manager

Une balise est une étiquette que vous affectez à une ressource AWS. Chaque balise est constituée d'une clé et d'une valeur que vous définissez toutes deux.

Les balises vous permettent de classer vos ressources AWS de différentes manières, par exemple, par objectif, par propriétaire ou par environnement. Par exemple, si vous souhaitez organiser et gérer vos ressources selon qu'elles sont utilisées pour le développement ou la production, vous pouvez baliser certaines d'entre elles avec la clé `Environment` et la valeur`Production`. Vous pouvez ensuite effectuer différents types de requêtes pour les ressources balisées`"Key=Environment,Values=Production"`. Par exemple, vous pouvez définir un ensemble de balises pour les instances gérées de votre compte qui vous aident à suivre ou à cibler les instances par système d'exploitation et environnement, telles que votre SUSE Linux Enterprise Server regroupé en `development` , et `staging` .`production` Vous pouvez également effectuer des opérations sur les ressources en spécifiant cette paire clé-valeur dans vos commandes, telles que l'exécution d'un script de mise à jour sur toutes les instances du groupe ou la vérification de l'état de ces instances.

Vous pouvez utiliser les balises appliquées à votre Systems Manager dans diverses opérations. Par exemple, vous pouvez cibler uniquement les instances gérées qui sont balisées avec une paire clé-valeur de balise spécifiée lorsque vous exécutez une commande (p. 946) ou affectez des cibles à une fenêtre de maintenance (p. 709). Vous pouvez également restreindre l'accès à vos ressources (p. 1304) en fonction des balises qui leur sont appliquées.

Pour aller plus loin, vous pouvez créer des groupes de ressources en spécifiant les mêmes balises pour les ressources AWS de différents types, pas seulement du même type. Après cela, vous pouvez utiliser les fonctionnalités Resource Groups Systems Manager pour afficher les informations relatives aux ressources au sein d'un groupe qui sont conformes et fonctionnent correctement, et à celles qui nécessitent une action. Les informations que vous affichez concernent tous les types de ressources AWS pouvant être ajoutés à un groupe de ressources, et ne concernent pas seulement les types de ressources Systems Manager pris en charge. Pour de plus amples informations, veuillez consulter Resource Groups dans AWS Systems Manager (p. 212) et Affichage des données d'opérations pour Groupes de ressources AWS (p. 214).

Le reste de ce chapitre décrit comment ajouter et supprimer des balises des ressources Systems Manager.

Rubriques

- Ressources Systems Manager pouvant être balisées (p. 1359)
- Balisage de documents Systems Manager (p. 1360)
- Balisage des fenêtres de maintenance (p. 1365)
- Balisage des instances gérées (p. 1369)
- BalisageOpsItems (p. 1374)
- Balisage des paramètres Systems Manager (p. 1377)
- Balisage des références de correctifs (p. 1381)

Ressources Systems Manager pouvant être balisées

Vous pouvez appliquer des balises aux types de ressources Systems Manager suivants.

- Documents

- Fenêtres de maintenance
- Instances gérées
- OpsItems
- Parameters
- Références de correctifs

À l'exception de ces types, chacun d'entre eux OpsItems peut être ajouté à un groupe de ressources.

Selon le type de ressource, vous pouvez utiliser des balises pour identifier les ressources à inclure dans une opération. Par exemple, vous pouvez baliser un groupe d'instances gérées, puis exécuter une tâche de fenêtre de maintenance qui cible uniquement les instances avec cette paire clé-valeur.

Vous pouvez également restreindre l'accès des utilisateurs à ces types de ressources en créant des stratégies IAM qui spécifient les balises auxquelles un utilisateur peut accéder et en attachant la stratégie à des comptes ou groupes d'utilisateurs. Voici quelques exemples de restriction de l'accès aux ressources à l'aide de balises.

- Vous pouvez appliquer une balise à un ensemble de documents Systems Manager personnalisés, puis créer une stratégie utilisateur qui accorde l'accès aux documents avec cette balise, mais pas aux autres (ou qui interdit l'accès à ces documents uniquement).
- Vous pouvez affecter des balises à OpsItems, puis créer des stratégies IAM qui limitent les utilisateurs ou groupes qui sont autorisés à afficher ou à mettre à jour ces ressources. Par exemple, les directeurs d'organisation pourraient bénéficier d'un accès total à tous les OpsItems , mais les développeurs de logiciels et les ingénieurs support pourraient uniquement bénéficier d'un accès aux projets ou segments de clients dont ils sont responsables.
- Vous pouvez appliquer une balise commune aux ressources des six types pris en charge et créer une stratégie IAM qui accorde l'accès uniquement à ces ressources, comme `Key=Project,Value=ProjectA` ou `Key=Environment,Value=Development` . Vous pouvez même accorder l'accès uniquement aux ressources auxquelles les deux paires de balises ont été affectées. Cela vous permet, par exemple, de limiter l'utilisation des utilisateurs uniquement avec les ressources pour ProjectA dans l'environnement de développement .

Vous pouvez utiliser la console Systems Manager Resource Groups, la console pour les types de ressources pris en charge (par exemple, la console Fenêtres de maintenance ou la console OpsCenter), l'AWS CLI, et Outils AWS pour PowerShell. Vous pouvez ajouter des balises lorsque vous créez ou mettez à jour une ressource. Par exemple, vous pouvez utiliser la commande add-tags-to-resource de l'AWS CLI pour ajouter des balises à n'importe quel type de ressource Systems Manager pris en charge après sa création. Vous pouvez utiliser la commande remove-tags-from-resource pour les supprimer.

Balisage de documents Systems Manager

Les rubriques de cette section décrivent l'utilisation de balises sur les documents Systems Manager.

Rubriques

- Création de documents avec des balises (p. 1360)
- Ajout de balises à des documents existants (p. 1361)
- Suppression de balises des documents Systems Manager (p. 1363)

Création de documents avec des balises

Vous pouvez ajouter des balises à des documents Systems Manager personnalisés au moment de leur création.

Pour plus d'informations, consultez les rubriques suivantes :

- Créer un document SSM (console) (p. 1269)
- Créer un document SSM (ligne de commande) (p. 1270)

Ajout de balises à des documents existants

Vous pouvez ajouter des balises aux documents Systems Manager personnalisés que vous possédez à l'aide de la console Systems Manager ou de la ligne de commande.

Rubriques

- Ajout de balises à un document SSM existant (console) (p. 1361)
- Ajout de balises à un document SSM existant (ligne de commande) (p. 1361)

Ajout de balises à un document SSM existant (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Documents dans le panneau de navigation.
3. Choisissez l'onglet M'appartenant.
4. Choisissez le nom du document auquel ajouter des balises, puis choisissez l'onglet Détails.
5. Dans la section Balises choisissez Modifier, puis ajoutez une ou plusieurs paires de balises clé-valeur.
6. Choisissez Save.

Ajout de balises à un document SSM existant (ligne de commande)

Pour ajouter des balises à un document existant (ligne de commande)

1. À l'aide de votre outil de ligne de commande préféré, exécutez la commande suivante pour afficher la liste des documents que vous pouvez baliser.

   Linux

   ```
   aws ssm list-documents
   ```

   Windows

   ```
   aws ssm list-documents
   ```

   PowerShell

   ```
   Get-SSMDocumentList
   ```

Notez le nom d'un document que vous voulez baliser.

2. Exécutez la commande suivante pour baliser un document.

   Linux

```
aws ssm add-tags-to-resource \
    --resource-type "Document" \
    --resource-id "document-name" \
    --tags "Key=tag-key,Value=tag-value"
```

   Windows

```
aws ssm add-tags-to-resource ^
    --resource-type "Document" ^
    --resource-id "document-name" ^
    --tags "Key=tag-key,Value=tag-value"
```

   PowerShell

```
$tag = New-Object Amazon.SimpleSystemsManagement.Model.Tag
```

```
$tag.Key = "tag-key"
```

```
$tag.Value = "tag-value"
```

```
Add-SSMResourceTag `
    -ResourceType "Document" `
    -ResourceId "document-name" `
    -Tag $tag `
    -Force
```

   tag-key est le nom d'une clé personnalisée que vous fournissez. Par exemple, région ou quartier.

   tag-value est le contenu personnalisé pour la valeur que vous souhaitez fournir pour cette clé. Par exemple, Ouestou Q321.

   document-name est le nom du SSM document que vous souhaitez baliser.

   En cas de réussite, la commande n'a aucune sortie.

3. Exécutez la commande suivante pour vérifier les balises du document :

   Linux

```
aws ssm list-tags-for-resource \
    --resource-type "Document" \
    --resource-id "document-name"
```

   Windows

```
aws ssm list-tags-for-resource ^
    --resource-type "Document" ^
```

```
    --resource-id "document-name"
```

PowerShell

```
Get-SSMResourceTag `
    -ResourceType "Document" `
    -ResourceId "document-name"
```

Suppression de balises des documents Systems Manager

Vous pouvez utiliser la console Systems Manager ou la ligne de commande pour supprimer des balises des documents Systems Manager.

Rubriques

- Suppression de balises des documents Systems Manager (console) (p. 1363)
- Suppression de balises des documents Systems Manager (ligne de commande) (p. 1363)

Suppression de balises des documents Systems Manager (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Documents dans le panneau de navigation.
3. Choisissez l'onglet M'appartenant.
4. Choisissez le nom du document duquel supprimer des balises, puis choisissez l'onglet Détails.
5. Dans la section Balises choisissez Modifier, puis Supprimer en regard de la paire de balises dont vous n'avez plus besoin.
6. Choisissez Save.

Suppression de balises des documents Systems Manager (ligne de commande)

1. À l'aide de votre outil de ligne de commande préféré, exécutez la commande suivante pour répertorier les documents de votre compte.

   Linux

   ```
   aws ssm list-documents
   ```

   Windows

   ```
   aws ssm list-documents
   ```

PowerShell

```
Get-SSMDocumentList
```

Notez le nom d'un document dont vous souhaitez supprimer des balises.

2. Exécutez la commande suivante pour supprimer des balises d'un document.

   Linux

```
aws ssm remove-tags-from-resource \
    --resource-type "Document" \
    --resource-id "document-name" \
    --tag-key "tag-key"
```

   Windows

```
aws ssm remove-tags-from-resource ^
    --resource-type "Document" ^
    --resource-id "document-name" ^
    --tag-key "tag-key"
```

   PowerShell

```
Remove-SSMResourceTag `
    -ResourceId "document-name" `
    -ResourceType "Document" `
    -TagKey "tag-key" `
    -Force
```

   document-name est le nom du SSM document duquel vous souhaitez supprimer des balises.

   tag-key est le nom d'une clé affectée au document. Par exemple, Environnement ou Quartier.

   En cas de réussite, la commande n'a aucune sortie.

3. Exécutez la commande suivante pour vérifier les balises du document :

   Linux

```
aws ssm list-tags-for-resource \
    --resource-type "Document" \
    --resource-id "document-name"
```

   Windows

```
aws ssm list-tags-for-resource ^
    --resource-type "Document" ^
    --resource-id "document-name"
```

   PowerShell

```
Get-SSMResourceTag `
    -ResourceType "Document" `
    -ResourceId "document-name"
```

Balisage des fenêtres de maintenance

Les rubriques de cette section décrivent l'utilisation des balises sur les fenêtres de maintenance.

Rubriques

- Création de fenêtres de maintenance avec des balises (p. 1365)
- Ajout de balises aux fenêtres de maintenance existantes (p. 1365)
- Suppression de balises des fenêtres de maintenance (p. 1367)

Création de fenêtres de maintenance avec des balises

Vous pouvez ajouter des balises aux fenêtres de maintenance au moment de leur création.

Pour plus d'informations, consultez les rubriques suivantes :

- Créer une fenêtre de maintenance (console) (p. 707)
- ??? (p. 717)Créer et configurer une fenêtre de maintenance (AWS CLI)

Ajout de balises aux fenêtres de maintenance existantes

Vous pouvez ajouter des balises aux fenêtres de maintenance que vous possédez à l'aide de la console Systems Manager ou de la ligne de commande.

Rubriques

- Ajout de balises à une fenêtre de maintenance existante (console) (p. 1365)
- Ajout de balises à une fenêtre de maintenance existante (AWS CLI) (p. 1366)
- Baliser une fenêtre de maintenance (Outils AWS pour PowerShell) (p. 1367)

Ajout de balises à une fenêtre de maintenance existante (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fenêtres de maintenance.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Fenêtres de maintenance.
3. Choisissez le nom d'une fenêtre de maintenance que vous avez déjà créée, puis choisissez les onglets Balises.
4. Choisissez Modification des balises puis Ajouter une balise.
5. Pour Clé, entrez une clé pour la balise, par exemple **`Environment`**.
6. (Facultatif) Dans Valeur, entrez une valeur pour la balise, par exemple **`Test`**.
7. Sélectionnez Save Changes (Enregistrer les modifications).

Ajout de balises à une fenêtre de maintenance existante (AWS CLI)

1. À l'aide de votre outil de ligne de commande préféré, exécutez la commande suivante pour afficher la liste des fenêtres de maintenance que vous pouvez baliser.

```
aws ssm describe-maintenance-windows
```

   Notez l'ID d'une fenêtre de maintenance que vous souhaitez baliser.

2. Exécutez la commande suivante pour baliser une fenêtre de maintenance.

   Linux

```
aws ssm add-tags-to-resource \
    --resource-type "MaintenanceWindow" \
    --resource-id "window-id" \
    --tags "Key=tag-key,Value=tag-value"
```

   Windows

```
aws ssm add-tags-to-resource ^
    --resource-type "MaintenanceWindow" ^
    --resource-id "window-id" ^
    --tags "Key=tag-key,Value=tag-value"
```

   En cas de réussite, la commande n'a aucune sortie.

   windows-id est l'ID de la fenêtre de maintenance que vous souhaitez baliser, par exemple mw-0c50858d01EXAMPLE .

   tag-key est le nom d'une clé personnalisée que vous fournissez. Par exemple, Environnement ou Projet.

   tag-value est le contenu personnalisé pour la valeur que vous souhaitez fournir pour cette clé. Par exemple, Production ou Q321.

3. Exécutez la commande suivante pour vérifier les balises de la fenêtre de maintenance.

   Linux

```
aws ssm list-tags-for-resource \
    --resource-type "MaintenanceWindow" \
    --resource-id "window-id"
```

   Windows

```
aws ssm list-tags-for-resource ^
    --resource-type "MaintenanceWindow" ^
    --resource-id "window-id"
```

Baliser une fenêtre de maintenance (Outils AWS pour PowerShell)

1. Exécutez la commande suivante pour répertorier les fenêtres de maintenance que vous pouvez baliser.

```
Get-SSMMaintenanceWindow
```

2. Exécutez les commandes suivantes pour baliser une fenêtre de maintenance.

```
$tag = New-Object Amazon.SimpleSystemsManagement.Model.Tag
```

```
$tag.Key = "tag-key"
```

```
$tag.Value = "tag-value"
```

```
Add-SSMResourceTag `
    -ResourceType "MaintenanceWindow" `
    -ResourceId "window-id" `
    -Tag $tag
```

   window-id l'ID de la fenêtre de maintenance que vous souhaitez baliser.

   tag-key est le nom d'une clé personnalisée que vous fournissez. Par exemple, Environnement ou Projet.

   tag-value est le contenu personnalisé pour la valeur que vous souhaitez fournir pour cette clé. Par exemple, Production ou Q321.

3. Exécutez la commande suivante pour vérifier les balises de la fenêtre de maintenance.

```
Get-SSMResourceTag `
    -ResourceType "MaintenanceWindow" `
    -ResourceId "window-id"
```

Suppression de balises des fenêtres de maintenance

Vous pouvez utiliser la console Systems Manager ou la ligne de commande pour supprimer des balises des fenêtres de maintenance.

Rubriques

- Suppression de balises des fenêtres de maintenance (console) (p. 1367)
- Suppression de balises des fenêtres de maintenance (ligne de commande) (p. 1368)

Suppression de balises des fenêtres de maintenance (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Fenêtres de maintenance.

   -ou-

Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Fenêtres de maintenance.

3. Choisissez le nom de la fenêtre de maintenance de laquelle supprimer des balises, puis choisissez l'onglet Balises.
4. Choisissez Modification des balises, puis choisissez Supprimer la balise en regard de la paire de balises dont vous n'avez plus besoin.
5. Sélectionnez Save Changes (Enregistrer les modifications).

Suppression de balises des fenêtres de maintenance (ligne de commande)

1. À l'aide de votre outil de ligne de commande préféré, exécutez la commande suivante pour répertorier les fenêtres de maintenance de votre compte.

   Linux

   ```
   aws ssm describe-maintenance-windows
   ```

   Windows

   ```
   aws ssm describe-maintenance-windows
   ```

   PowerShell

   ```
   Get-SSMMaintenanceWindows
   ```

   Notez l'ID d'une fenêtre de maintenance dont vous souhaitez supprimer des balises.

2. Exécutez la commande suivante pour supprimer des balises d'une fenêtre de maintenance.

   Linux

   ```
   aws ssm remove-tags-from-resource \
       --resource-type "MaintenanceWindow" \
       --resource-id "window-id" \
       --tag-key "tag-key"
   ```

   Windows

   ```
   aws ssm remove-tags-from-resource ^
       --resource-type "MaintenanceWindow" ^
       --resource-id "window-id" ^
       --tag-key "tag-key"
   ```

   PowerShell

   ```
   Remove-SSMResourceTag `
       -ResourceType "MaintenanceWindow" `
       -ResourceId "window-id" `
       -TagKey "tag-key
   ```

windows-id est l'ID de la fenêtre de maintenance de laquelle vous souhaitez supprimer une balise, par exemple mw-0c50858d01EXAMPLE .

tag-key est le nom d'une clé affectée à la fenêtre de maintenance. Par exemple, Environnement ou Quartier.

En cas de réussite, la commande n'a aucune sortie.

3. Exécutez la commande suivante pour vérifier les balises de la fenêtre de maintenance.

   Linux

```
aws ssm list-tags-for-resource \
    --resource-type "MaintenanceWindow" \
    --resource-id "window-id"
```

   Windows

```
aws ssm list-tags-for-resource ^
    --resource-type "MaintenanceWindow" ^
    --resource-id "window-id"
```

   PowerShell

```
Get-SSMResourceTag `
    -ResourceType "MaintenanceWindow" `
    -ResourceId "window-id"
```

Balisage des instances gérées

Les rubriques de cette section décrivent l'utilisation des balises sur les instances gérées.

Une instance gérée est une machine configurée pour AWS Systems Manager. Il s'agit notamment des instances EC2, ainsi que des serveurs ou des machines virtuelles (VM) sur site que vous avez configurés afin qu'ils soient gérés par Systems Manager dans un environnement hybride.

Les instructions de cette rubrique s'appliquent à toute machine gérée à l'aide de Systems Manager.

Rubriques

- Création ou activation d'instances gérées avec des balises (p. 1369)
- Ajout de balises à des instances gérées existantes (p. 1370)
- Suppression de balises des instances gérées (p. 1372)

Création ou activation d'instances gérées avec des balises

Vous pouvez ajouter des balises aux instances EC2 au moment de leur création. Vous pouvez ajouter des balises aux serveurs et aux machines virtuelles (VM) sur site au moment de leur activation.

Pour plus d'informations, consultez les rubriques suivantes :

- Pour les instances EC2, veuillez consulter Balisage de vos ressources Amazon EC2 dans le Amazon EC2 Guide de l'utilisateur pour les instances Linux. (Le contenu s'applique aux instances EC2 pour Linux et pour Windows)
- Pour les serveurs sur site et VMs , consultez Créer une activation d'instance gérée pour un environnement hybride (p. 55) .

Ajout de balises à des instances gérées existantes

Vous pouvez ajouter des balises aux instances gérées à l'aide de la console Systems Manager ou de la ligne de commande.

Rubriques

- Ajout de balises à une instance gérée existante (console) (p. 1370)
- Ajout de balises à une instance gérée existante (ligne de commande) (p. 1370)

Ajout de balises à une instance gérée existante (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Instances gérées.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Instances gérées.
3. Choisissez le nom de l'instance gérée à laquelle ajouter des balises, puis choisissez l'onglet Balises.

   Note

   Si une instance Amazon EC2 que vous vous attendez à voir n'est pas répertoriée, consultez Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 865) pour obtenir des conseils de dépannage.
4. Dans la section Balises choisissez Modifier, puis ajoutez une ou plusieurs paires de balises clé-valeur.
5. Choisissez Save.

Ajout de balises à une instance gérée existante (ligne de commande)

Pour ajouter des balises à une instance gérée existante (ligne de commande)

1. À l'aide de votre outil de ligne de commande préféré, exécutez la commande suivante pour afficher la liste des instances gérées que vous pouvez baliser.

   Linux

   ```
   aws ssm describe-instance-information
   ```

   Windows

   ```
   aws ssm describe-instance-information
   ```

PowerShell

```
Get-SSMInstanceInformation
```

Notez l'ID d'une instance gérée que vous souhaitez baliser.

> **Note**
>
> Les machines qui ont été enregistrées pour être utilisées avec Systems Manager dans un environnement hybride commencent par `mi-` , par exemple `mi-0471e04240EXAMPLE;` . Les instances EC2 IDs commencent par `i-` , par exemple `i-02573cafcfEXAMPLE` .

2. Exécutez la commande suivante pour baliser une instance gérée.

   Linux

   ```
   aws ssm add-tags-to-resource \
       --resource-type "ManagedInstance" \
       --resource-id "instance-id" \
       --tags "Key=tag-key,Value=tag-value"
   ```

   Windows

   ```
   aws ssm add-tags-to-resource ^
       --resource-type "ManagedInstance" ^
       --resource-id "instance-id" ^
       --tags "Key=tag-key,Value=tag-value"
   ```

   PowerShell

   ```
   $tag = New-Object Amazon.SimpleSystemsManagement.Model.Tag
   ```

   ```
   $tag.Key = "tag-key"
   ```

   ```
   $tag.Value = "tag-value"
   ```

   ```
   Add-SSMResourceTag `
       -ResourceType "ManagedInstance" `
       -ResourceId "instance-id" `
       -Tag $tag `
       -Force
   ```

   `tag-key` est le nom d'une clé personnalisée que vous fournissez. Par exemple, région ou quartier.

   `tag-value` est le contenu personnalisé pour la valeur que vous souhaitez fournir pour cette clé. Par exemple, Ouestou Q321.

   `instance-id` correspond à l'ID de l'instance gérée que vous souhaitez baliser.

   En cas de réussite, la commande n'a aucune sortie.

3. Exécutez la commande suivante pour vérifier les balises de l'instance gérée.

Linux

```
aws ssm list-tags-for-resource \
    --resource-type "ManagedInstance" \
    --resource-id "instance-id"
```

Windows

```
aws ssm list-tags-for-resource ^
    --resource-type "ManagedInstance" ^
    --resource-id "instance-id"
```

PowerShell

```
Get-SSMResourceTag `
    -ResourceType "ManagedInstance" `
    -ResourceId "instance-id"
```

Suppression de balises des instances gérées

Vous pouvez utiliser la console Systems Manager ou la ligne de commande pour supprimer les balises des instances gérées.

Rubriques

- Suppression de balises des instances gérées (console) (p. 1372)
- Suppression de balises des instances gérées (ligne de commande) (p. 1372)

Suppression de balises des instances gérées (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Instances gérées.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Instances gérées.
3. Choisissez le nom de l'instance gérée de laquelle supprimer les balises, puis choisissez l'onglet Balises.
4. Dans la section Balises choisissez Modifier, puis Supprimer en regard de la paire de balises dont vous n'avez plus besoin.
5. Choisissez Save.

Suppression de balises des instances gérées (ligne de commande)

1. À l'aide de votre outil de ligne de commande préféré, exécutez la commande suivante pour répertorier les instances gérées dans votre compte.

Linux

```
aws ssm describe-instance-information
```

Windows

```
aws ssm describe-instance-information
```

PowerShell

```
Get-SSMInstanceInformation
```

Notez le nom d'une instance gérée dont vous souhaitez supprimer les balises.

2. Exécutez la commande suivante pour supprimer des balises d'une instance gérée.

   Linux

   ```
   aws ssm remove-tags-from-resource \
       --resource-type "ManagedInstance" \
       --resource-id "instance-id" \
       --tag-key "tag-key"
   ```

   Windows

   ```
   aws ssm remove-tags-from-resource ^
       --resource-type "ManagedInstance" ^
       --resource-id "instance-id" ^
       --tag-key "tag-key"
   ```

   PowerShell

   ```
   Remove-SSMResourceTag `
       -ResourceId "instance-id" `
       -ResourceType "ManagedInstance" `
       -TagKey "tag-key" `
       -Force
   ```

   instance-id est le nom de l'instance gérée dont vous souhaitez supprimer des balises.

   tag-key est le nom d'une clé attribuée à l'instance gérée. Par exemple, Environnement ou Quartier.

   En cas de réussite, la commande n'a aucune sortie.

3. Exécutez la commande suivante pour vérifier les balises de l'instance gérée.

   Linux

   ```
   aws ssm list-tags-for-resource \
       --resource-type "ManagedInstance" \
       --resource-id "instance-id"
   ```

Windows

```
aws ssm list-tags-for-resource ^
    --resource-type "ManagedInstance" ^
    --resource-id "instance-id"
```

PowerShell

```
Get-SSMResourceTag `
    -ResourceType "ManagedInstance" `
    -ResourceId "instance-id"
```

BalisageOpsItems

Les rubriques de cette section décrivent l'utilisation des balises sur OpsItems .

Rubriques

- Création avec des balisesOpsItems (p. 1374)
- Ajout de balises à desOpsItems (p. 1374)
- Suppression de balises d'Systems ManagerOpsItems (p. 1376)

Création avec des balisesOpsItems

Vous pouvez ajouter des balises aux personnalisées Systems Manager OpsItems au moment de leur création si vous utilisez un outil de ligne de commande.

Pour de plus amples informations, veuillez consulter la rubrique suivante :

- Pour créer manuellement un OpsItem (AWS CLI) (p. 190)

Ajout de balises à desOpsItems

Vous pouvez ajouter des balises à à l'aide OpsItems d'un outil de ligne de commande.

Rubriques

- Ajout de balises à un existant OpsItem (ligne de commande) (p. 1374)

Ajout de balises à un existant OpsItem (ligne de commande)

Pour ajouter des balises à un existant OpsItem (ligne de commande)

1. À l'aide de votre outil de ligne de commande préféré, exécutez la commande suivante pour afficher la liste des OpsItem que vous pouvez baliser.

   Linux

   ```
   aws ssm describe-ops-items
   ```

Windows

```
aws ssm describe-ops-items
```

PowerShell

```
Get-SSMOpsItemSummary
```

Notez l'ID d'un OpsItem que vous souhaitez baliser.

2. Exécutez la commande suivante pour baliser un OpsItem .

Linux

```
aws ssm add-tags-to-resource \
    --resource-type "OpsItem" \
    --resource-id "ops-item-id" \
    --tags "Key=tag-key,Value=tag-value"
```

Windows

```
aws ssm add-tags-to-resource ^
    --resource-type "OpsItem" ^
    --resource-id "ops-item-id" ^
    --tags "Key=tag-key,Value=tag-value"
```

PowerShell

```
$tag = New-Object Amazon.SimpleSystemsManagement.Model.Tag
```

```
$tag.Key = "tag-key"
```

```
$tag.Value = "tag-value"
```

```
Add-SSMResourceTag `
    -ResourceType "OpsItem" `
    -ResourceId "ops-item-id" `
    -Tag $tag `
    -Force
```

tag-key est le nom d'une clé personnalisée que vous fournissez. Par exemple, région ou quartier.

tag-value est le contenu personnalisé pour la valeur que vous souhaitez fournir pour cette clé. Par exemple, Ouestou Q321.

ops-item-id est l'ID du que OpsItem vous souhaitez baliser.

En cas de réussite, la commande n'a aucune sortie.

3. Exécutez la commande suivante pour vérifier les OpsItem balises.

Linux

```
aws ssm list-tags-for-resource \
    --resource-type "OpsItem" \
    --resource-id "ops-item-id"
```

Windows

```
aws ssm list-tags-for-resource ^
    --resource-type "OpsItem" ^
    --resource-id "ops-item-id"
```

PowerShell

```
Get-SSMResourceTag `
    -ResourceType "OpsItem" `
    -ResourceId "ops-item-id"
```

Suppression de balises d'Systems ManagerOpsItems

Vous pouvez utiliser un outil de ligne de commande pour supprimer des balises d' Systems Manager OpsItems .

Rubriques

- Suppression de balises d' OpsItems (ligne de commande) (p. 1376)

Suppression de balises d' OpsItems (ligne de commande)

1. À l'aide de votre outil de ligne de commande préféré, exécutez la commande suivante pour répertorier les instances de votre compte.OpsItems

   Linux

   ```
   aws ssm describe-ops-items
   ```

   Windows

   ```
   aws ssm describe-ops-items
   ```

   PowerShell

   ```
   Get-SSMOpsItemSummary
   ```

   Notez le nom d'un OpsItem duquel vous souhaitez supprimer des balises.
2. Exécutez la commande suivante pour supprimer des balises d'un OpsItem .

   Linux

   ```
   aws ssm remove-tags-from-resource \
       --resource-type "OpsItem" \
   ```

```
    --resource-id "ops-item-id" \
    --tag-key "tag-key"
```

Windows

```
aws ssm remove-tags-from-resource ^
    --resource-type "OpsItem" ^
    --resource-id "ops-item-id" ^
    --tag-key "tag-key"
```

PowerShell

```
Remove-SSMResourceTag `
    -ResourceId "ops-item-id" `
    -ResourceType "OpsItem" `
    -TagKey "tag-key" `
    -Force
```

ops-item-id est l'ID du de laquelle vous souhaitez supprimer des balises.OpsItem

tag-key est le nom d'une clé affectée au OpsItem . Par exemple, l'environnement ou le trimestre.

En cas de réussite, la commande n'a aucune sortie.

3. Exécutez la commande suivante pour vérifier les OpsItem balises.

Linux

```
aws ssm list-tags-for-resource \
    --resource-type "OpsItem" \
    --resource-id "ops-item-id"
```

Windows

```
aws ssm list-tags-for-resource ^
    --resource-type "OpsItem" ^
    --resource-id "ops-item-id"
```

PowerShell

```
Get-SSMResourceTag `
    -ResourceType "OpsItem" `
    -ResourceId "ops-item-id"
```

Balisage des paramètres Systems Manager

Les rubriques de cette section décrivent l'utilisation des balises sur les paramètres Systems Manager.

Rubriques

- Création de paramètres avec des balises (p. 1378)
- Ajout de balises aux paramètres existants (p. 1378)
- Suppression de balises des paramètres Systems Manager (p. 1380)

Création de paramètres avec des balises

Vous pouvez ajouter des balises aux paramètres Systems Manager au moment de leur création.

Pour plus d'informations, consultez les rubriques suivantes :

- Créer un paramètre (console) (p. 245)
- Créer un paramètre Systems Manager (AWS CLI) (p. 246)
- Créer un paramètre Systems Manager (Outils pour Windows PowerShell) (p. 256)

Ajout de balises aux paramètres existants

Vous pouvez ajouter des balises aux paramètres Systems Manager personnalisés que vous possédez à l'aide de la console Systems Manager ou de la ligne de commande.

Rubriques

- Ajout de balises à un paramètre existant (console) (p. 1378)
- Ajout de balises à un paramètre existant (AWS CLI) (p. 1378)
- Ajout de balises à un paramètre existant (Outils AWS pour PowerShell) (p. 1379)

Ajout de balises à un paramètre existant (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Parameter Store.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Parameter Store.
3. Choisissez le nom d'un paramètre que vous avez déjà créé, puis choisissez l'onglet Balises.
4. Dans la première zone, entrez une clé pour la balise, telle que **`Environment`**.
5. Dans la deuxième zone, entrez une valeur pour la balise, telle que **`Test`**.
6. Choisissez Save.

Ajout de balises à un paramètre existant (AWS CLI)

1. À l'aide de votre outil de ligne de commande préféré, exécutez la commande suivante pour afficher la liste des paramètres que vous pouvez baliser.

   ```
   aws ssm describe-parameters
   ```

   Notez le nom d'un paramètre auquel vous voulez attribuer une balise.
2. Exécutez la commande suivante pour attribuer une balise à un paramètre.

   ```
   aws ssm add-tags-to-resource --resource-type "Parameter" --resource-id "parameter-name"
    --tags "Key=tag-key,Value=tag-value"
   ```

   En cas de réussite, la commande n'a aucune sortie.

parameter-name est le nom du SSM paramètre que vous souhaitez baliser.

tag-key est le nom d'une clé personnalisée que vous fournissez. Par exemple, Environnement ou Projet.

tag-value est le contenu personnalisé pour la valeur que vous souhaitez fournir pour cette clé. Par exemple, Production ou Q321.

3. Exécutez la commande suivante pour vérifier les balises du paramètre.

```
aws ssm list-tags-for-resource --resource-type "Parameter" --resource-id "parameter-name"
```

Ajout de balises à un paramètre existant (Outils AWS pour PowerShell)

1. Exécutez la commande suivante pour établir une liste des paramètres auxquels vous pouvez attribuer une balise.

```
Get-SSMParameterList
```

2. Exécutez les commandes suivantes pour attribuer une balise à un paramètre.

```
$tag = New-Object Amazon.SimpleSystemsManagement.Model.Tag
```

```
$tag.Key = "tag-key"
```

```
$tag.Value = "tag-value"
```

```
Add-SSMResourceTag `
    -ResourceType "Parameter" `
    -ResourceId "parameter-name" `
    -Tag $tag `
    -Force
```

parameter-name le nom du SSM paramètre que vous souhaitez baliser.

tag-key est le nom d'une clé personnalisée que vous fournissez. Par exemple, Environnement ou Projet.

tag-value est le contenu personnalisé pour la valeur que vous souhaitez fournir pour cette clé. Par exemple, Production ou Q321.

3. Exécutez la commande suivante pour vérifier les balises du paramètre.

```
Get-SSMResourceTag `
    -ResourceType "Parameter" `
    -ResourceId "parameter-name"
```

Suppression de balises des paramètres Systems Manager

Vous pouvez utiliser la console Systems Manager ou la ligne de commande pour supprimer des balises des paramètres Systems Manager.

Rubriques

- Suppression de balises des paramètres Systems Manager (console) (p. 1380)
- Suppression de balises des paramètres Systems Manager (ligne de commande) (p. 1380)

Suppression de balises des paramètres Systems Manager (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, choisissez Parameter Store.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Parameter Store.
3. Choisissez le nom du paramètre duquel supprimer des balises, puis choisissez l'onglet Balises.
4. Choisissez Supprimer en regard de la paire de balises dont vous n'avez plus besoin.
5. Choisissez Save.

Suppression de balises des paramètres Systems Manager (ligne de commande)

1. À l'aide de votre outil de ligne de commande préféré, exécutez la commande suivante pour répertorier les paramètres de votre compte.

   Linux

   ```
   aws ssm describe-parameters
   ```

   Windows

   ```
   aws ssm describe-parameters
   ```

   PowerShell

   ```
   Get-SSMParameterList
   ```

   Notez le nom d'un paramètre dont vous souhaitez supprimer les balises.
2. Exécutez la commande suivante pour supprimer les balises d'un paramètre.

Linux

```
aws ssm remove-tags-from-resource \
    --resource-type "Parameter" \
    --resource-id "parameter-name" \
    --tagkey "tag-key"
```

Windows

```
aws ssm remove-tags-from-resource ^
    --resource-type "Parameter" ^
    --resource-id "parameter-name" ^
    --tag-key "tag-key"
```

PowerShell

```
Remove-SSMResourceTag
    -ResourceId "parameter-name"
    -ResourceType "Parameter"
    -TagKey "tag-key
```

parameter-name est le nom du SSM paramètre duquel vous souhaitez supprimer une balise.

tag-key est le nom d'une clé affectée au paramètre. Par exemple, Environnement ou Quartier.

En cas de réussite, la commande n'a aucune sortie.

3. Exécutez la commande suivante pour vérifier les balises du document :

Linux

```
aws ssm list-tags-for-resource \
    --resource-type "Parameter" \
    --resource-id "parameter-name"
```

Windows

```
aws ssm list-tags-for-resource ^
    --resource-type "Parameter" ^
    --resource-id "parameter-name"
```

PowerShell

```
Get-SSMResourceTag `
    -ResourceType "Parameter" `
    -ResourceId "parameter-name"
```

Balisage des références de correctifs

Les rubriques de cette section décrivent l'utilisation des balises sur les références de correctifs.

Rubriques

- Création de références de correctifs avec des balises (p. 1382)

- Ajout de balises aux références de correctifs existantes (p. 1382)
- Suppression de balises des références de correctifs (p. 1384)

Création de références de correctifs avec des balises

Vous pouvez ajouter des balises aux paramètres Systems Manager au moment de leur création.

Pour plus d'informations, consultez les rubriques suivantes :

- Utilisation de références de correctifs personnalisées (p. 1127)
- Créer une référence de correctifs (p. 1148)
- Création d'une référence de correctifs avec des référentiels personnalisés pour les différentes versions du système d'exploitation (p. 1149)

Ajout de balises aux références de correctifs existantes

Vous pouvez ajouter des balises aux références de correctifs que vous possédez à l'aide de la console Systems Manager ou de la ligne de commande.

Rubriques

- Ajout de balises à une référence de correctifs existante (console) (p. 1382)
- Ajout de balises à un paramètre existant (AWS CLI) (p. 1382)
- Baliser une référence de correctifs (Outils AWS pour PowerShell) (p. 1383)

Ajout de balises à une référence de correctifs existante (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le volet de navigation, choisissez Gestionnaire de correctifs.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Gestionnaire de correctifs.
3. Choisissez le nom d'une référence de correctifs personnalisée que vous avez déjà créée, faites défiler jusqu'à la section Tableau des balises, puis choisissez Modification des balises.
4. Sélectionnez Add Tags (Ajouter des balises).
5. Pour Clé, entrez une clé pour la balise, par exemple **`Environment`**.
6. (Facultatif) Dans Valeur, entrez une valeur pour la balise, par exemple **`Test`**.
7. Sélectionnez Save Changes (Enregistrer les modifications).

Ajout de balises à un paramètre existant (AWS CLI)

1. À l'aide de votre outil de ligne de commande préféré, exécutez la commande suivante pour afficher la liste des paramètres que vous pouvez baliser.

```
aws ssm describe-patch-baselines
```

Notez l'ID d'une référence de correctifs que vous souhaitez baliser.

2. Exécutez la commande suivante pour attribuer une balise à une référence de correctifs.

   Linux

   ```
   aws ssm add-tags-to-resource \
       --resource-type "PatchBaseline" \
       --resource-id "baseline-id" \
       --tags "Key=tag-key,Value=tag-value"
   ```

   Windows

   ```
   aws ssm add-tags-to-resource ^
       --resource-type "PatchBaseline" ^
       --resource-id "baseline-id" ^
       --tags "Key=tag-key,Value=tag-value"
   ```

   En cas de réussite, la commande n'a aucune sortie.

   baseline-id est l'ID de la référence de correctifs que vous souhaitez baliser, par exemple pb-0c10e65780EXAMPLE .

   tag-key est le nom d'une clé personnalisée que vous fournissez. Par exemple, Environnement ou Projet.

   tag-value est le contenu personnalisé pour la valeur que vous souhaitez fournir pour cette clé. Par exemple, Production ou Q321.

3. Exécutez la commande suivante pour vérifier les balises de la référence de correctifs.

   Linux

   ```
   aws ssm list-tags-for-resource \
       --resource-type "PatchBaseline" \
       --resource-id "baseline-id"
   ```

   Windows

   ```
   aws ssm list-tags-for-resource ^
       --resource-type "PatchBaseline" ^
       --resource-id "patchbaseline-id"
   ```

Baliser une référence de correctifs (Outils AWS pour PowerShell)

1. Exécutez la commande suivante pour établir une liste des références de correctifs auxquelles vous pouvez attribuer une balise.

   ```
   Get-SSMPatchBaseline
   ```

2. Exécutez les commandes suivantes pour attribuer une balise à une référence de correctifs.

   ```
   $tag = New-Object Amazon.SimpleSystemsManagement.Model.Tag
   ```

```
$tag.Key = "tag-key"
```

```
$tag.Value = "tag-value"
```

```
Add-SSMResourceTag `
    -ResourceType "PatchBaseline" `
    -ResourceId "baseline-id" `
    -Tag $tag `
    -Force
```

patch-baseline-name le nom du SSM paramètre que vous souhaitez baliser.

tag-key est le nom d'une clé personnalisée que vous fournissez. Par exemple, Environnement ou Projet.

tag-value est le contenu personnalisé pour la valeur que vous souhaitez fournir pour cette clé. Par exemple, Production ou Q321.

3. Exécutez la commande suivante pour vérifier les balises de la référence de correctifs.

```
Get-SSMResourceTag `
    -ResourceType "PatchBaseline" `
    -ResourceId "baseline-id"
```

Suppression de balises des références de correctifs

Vous pouvez utiliser la console Systems Manager ou la ligne de commande pour supprimer les balises de la références de correctifs.

Rubriques

- Suppression de balises de la références de correctifs (console) (p. 1384)
- Suppression de balises des références de correctifs (ligne de commande) (p. 1385)

Suppression de balises de la références de correctifs (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le volet de navigation, choisissez Gestionnaire de correctifs.

   -ou-

   Si la page d'accueil AWS Systems Manager s'ouvre en premier, choisissez l'icône de menu (≡) pour ouvrir le panneau de navigation, puis choisissez Gestionnaire de correctifs.
3. Choisissez le nom de la référence de correctifs de laquelle supprimer des balises, faites défiler vers le bas jusqu'à la section Tableau des balises, puis choisissez l'onglet Modification des balises.
4. Choisissez Supprimer la balise en regard de la paire de balises dont vous n'avez plus besoin.
5. Sélectionnez Save Changes (Enregistrer les modifications).

Suppression de balises des références de correctifs (ligne de commande)

1. À l'aide de votre outil de ligne de commande préféré, exécutez la commande suivante pour répertorier les références de correctifs dans votre compte.

   Linux

   ```
   aws ssm describe-patch-baselines
   ```

   Windows

   ```
   aws ssm describe-patch-baselines
   ```

   PowerShell

   ```
   Get-SSMPatchBaseline
   ```

   Notez l'ID d'une référence de correctifs dont vous souhaitez supprimer des balises.

2. Exécutez la commande suivante pour supprimer des balises d'une référence de correctifs.

   Linux

   ```
   aws ssm remove-tags-from-resource \
       --resource-type "PatchBaseline" \
       --resource-id "baseline-id" \
       --tag-key "tag-key"
   ```

   Windows

   ```
   aws ssm remove-tags-from-resource ^
       --resource-type "PatchBaseline" ^
       --resource-id "baseline-id" ^
       --tag-key "tag-key"
   ```

   PowerShell

   ```
   Remove-SSMResourceTag `
       -ResourceType "PatchBaseline" `
       -ResourceId "baseline-id" `
       -TagKey "tag-key
   ```

   baseline-id est l'ID de la référence de correctifs que vous souhaitez baliser, par exemple pb-0c10e65780EXAMPLE .

   tag-key est le nom d'une clé affectée à la référence de correctifs. Par exemple, Environnement ou Quartier.

   En cas de réussite, la commande n'a aucune sortie.

3. Exécutez la commande suivante pour vérifier les balises de la référence de correctifs.

Linux

```
aws ssm list-tags-for-resource \
    --resource-type "PatchBaseline" \
    --resource-id "baseline-id"
```

Windows

```
aws ssm list-tags-for-resource ^
    --resource-type "PatchBaseline" ^
    --resource-id "baseline-id"
```

PowerShell

```
Get-SSMResourceTag `
    -ResourceType "PatchBaseline" `
    -ResourceId "baseline-id"
```

Référence AWS Systems Manager

Les informations et les rubriques suivantes peuvent vous aider à mieux mettre en œuvre des solutions Systems Manager.

Principal

Dans AWS Identity and Access Management (IAM), vous pouvez accorder ou refuser à un service l'accès aux ressources en utilisant l'élément de stratégie Principal. La valeur de l'élément de stratégie Principal pour Systems Manager est `ssm.amazonaws.com`.

Régions et points de terminaison pris en charge

Veuillez consulter Points de terminaison de service Systems Manager dans le document Référence générale d'Amazon Web Services.

Quotas de service

Veuillez consulter Quotas de service Systems Manager dans le Référence générale d'Amazon Web Services..

API Reference

Veuillez consulter Référence d'API AWS Systems Manager.

AWS CLI Référence de la commande

Veuillez consulter AWS Systems Manager section of the AWS CLI Command Reference.

Outils AWS pour PowerShell Référence de l'applet de commande

Veuillez consulter AWS Systems Manager section of the Outils AWS pour PowerShell Cmdlet Reference.

Agent SSM Référentiel sur GitHub

Consultez aws/amazon-ssm-agent.

Poser une question

AWS Systems Manager Forum des développeurs

Blog des actualités AWS

Outils de gestion

Autres sujets de référence

- Référence : modèles et types Amazon EventBridge d'événements pourSystems Manager (p. 1387)
- Référence : Expressions cron et rate pour Systems Manager (p. 1392)
- Référence : ec2messages, ssmmessages et autres appels d'API (p. 1399)
- Référence : Création de chaînes de date et d'heure formatées pour Systems Manager (p. 1400)

Référence : modèles et types Amazon EventBridge d'événements pourSystems Manager

> **Note**
>
> Amazon EventBridge est le meilleur moyen de gérer vos événements. CloudWatch Events et EventBridge sont les mêmes services sous-jacents et API, mais EventBridge fournit plus de

fonctions. Les modifications que vous apportez à CloudWatch ou à EventBridge apparaîtront dans chaque console. Pour de plus amples informations, veuillez consulter Guide de l'utilisateur Amazon EventBridge.

Amazon EventBridgeAvec (EventBridge), vous pouvez créer des règles qui correspondent à des événements entrants et les acheminer vers des cibles pour traitement.

Un événement indique un changement dans un environnement dans vos propres applications, applications SaaS (logiciel en tant que service) ou un AWS service . Une fois qu'un type d'événement spécifié dans une règle est détecté, l'EventBridgeachemine vers une cible spécifiée pour traitement. Les cibles peuvent inclure des instances Amazon Elastic Compute Cloud (Amazon EC2), AWS Lambda des fonctions, des Amazon Kinesis flux, des tâches Amazon Elastic Container Service (Amazon ECS), des machines dAWS Step Functions'état, des rubriques Amazon Simple Notification Service (Amazon SNS), des files d'attente Amazon Simple Queue Service (Amazon SQS), des cibles intégrées, etc.

Pour plus d'informations sur la création EventBridge de règles, consultez les rubriques suivantes :

- Surveillance des événements Systems Manager avec Amazon EventBridge (p. 1340)
- AWS Systems ManagerÉvénements dans le Guide de l'utilisateur Amazon EventBridge .
- Mise en route avec Amazon EventBridge dans leGuide de l'utilisateur Amazon EventBridge

Le reste de cette rubrique décrit les types d'Systems Managerévénements que vous pouvez inclure dans vos EventBridge règles.

Type d'événement : Automation

Nom du type d'événement	Description des événements que vous pouvez ajouter à une règle
Notification de changement de statut d'exécution d'Automation EC2	L'état global d'un flux de travail Automation change.Vous pouvez ajouter une ou plusieurs des modifications de statut suivantes à une règle d'événement : • Annulée • Failed (Échec) • InProgress • En suspens • Success (Succès) • TimedOut
Notification de changement de statut d'étape d'automatisation EC2	L'état d'une étape spécifique dans un flux de travail Automation change.Vous pouvez ajouter une ou plusieurs des modifications de statut suivantes à une règle d'événement : • Annulée • Failed (Échec) • InProgress • En suspens • Success (Succès) • TimedOut

Type d'événement : Conformité de la configuration

Nom du type d'événement	Description des événements que vous pouvez ajouter à une règle
Changement d'état de conformité de configuration	L'état d'une instance gérée change soit pour la conformité de l'association, soit pour la conformité des correctifs.Vous pouvez ajouter une ou plusieurs des modifications d'état suivantes à une règle d'événement : • conforme • non_conforme

Type d'événement : Inventaire

Nom du type d'événement	Description des événements que vous pouvez ajouter à une règle
Changement d'état de ressource d'inventaire	État d'une ressource qui est suivie par les changements Systems Manager d'inventaire.

Type d'événement :State Manager

Nom du type d'événement	Description des événements que vous pouvez ajouter à une règle
Changements d'état d'association dans EC2 State Manager	L'état global d'une association change au fur et à mesure qu'elle est appliquée.Vous pouvez ajouter une ou plusieurs des modifications d'état suivantes à une règle d'événement : • Failed (Échec) • En suspens • Success (Succès)
Changements d'état d'association d'instance dans EC2 State Manager	L'état d'une instance gérée unique qui est ciblée par une modification d'association.Vous pouvez ajouter une ou plusieurs des modifications d'état suivantes à une règle d'événement : • Failed (Échec) • En suspens • Success (Succès)

Type d'événement : Fenêtre de maintenance

Nom du type d'événement	Description des événements que vous pouvez ajouter à une règle
Notification de changement de statut de fenêtre de maintenance	Le statut global d'une ou de plusieurs fenêtres de maintenance change.Vous pouvez ajouter une ou plusieurs des modifications d'état suivantes à une règle d'événement : • DISABLED • ENABLED
Notification d'enregistrement de la cible de la fenêtre de maintenance	Le statut d'une ou de plusieurs cibles de fenêtre de maintenance change.Vous pouvez ajouter une ou plusieurs des modifications d'état suivantes à une règle d'événement : • DEREGISTERED (OBSOLÈTE) • REGISTERED • MIS À JOUR
Notification de changement d'état d'exécution de la fenêtre de maintenance	Le statut global d'une fenêtre de maintenance change pendant qu'elle est en cours d'exécution.Vous pouvez ajouter une ou plusieurs des modifications d'état suivantes à une règle d'événement : • CANCELLED • ANNULATION • ÉCHEC • IN_PROGRESS • EN SUSPENS • RESTAURATION DE LA LANCEMENT • SUCCESS • SORTIE TIMED_OUT
Notification de changement d'état d'exécution de tâche de fenêtre de maintenance	L'état d'une tâche dans une fenêtre de maintenance change pendant son exécution.Vous pouvez ajouter une ou plusieurs des modifications d'état suivantes à une règle d'événement : • CANCELLED • ANNULATION • ÉCHEC • IN_PROGRESS • SUCCESS • SORTIE TIMED_OUT
Notification de changement d'état d'appel de la cible de la tâche de fenêtre de maintenance	L'état d'une tâche de fenêtre de maintenance sur une cible spécifique change.Vous pouvez ajouter une ou plusieurs des modifications d'état suivantes à une règle d'événement :

Nom du type d'événement	Description des événements que vous pouvez ajouter à une règle
	• CANCELLED • ANNULATION • ÉCHEC • IN_PROGRESS • SUCCESS • SORTIE TIMED_OUT
Notification d'enregistrement de la tâche de fenêtre de maintenance	L'état d'une ou de plusieurs tâches de fenêtre de maintenance change.Vous pouvez ajouter une ou plusieurs des modifications d'état suivantes à une règle d'événement : • DEREGISTERED (OBSOLÈTE) • REGISTERED • MIS À JOUR

Type d'événement :Parameter Store

Nom du type d'événement	Description des événements que vous pouvez ajouter à une règle
de modification du Parameter Store	L'état d'un paramètre change.Vous pouvez ajouter une ou plusieurs des modifications d'état suivantes à une règle d'événement : • Créer • Mise à jour • Supprimer • LabelParameterVersion Pour plus d'informations, consultez la section Configuration de EventBridge pour les paramètres (p. 239).
Action de stratégie Parameter Store	Une condition de modification avancée d'une stratégie de paramètre est remplie.Vous pouvez ajouter une ou plusieurs des modifications de statut suivantes à une règle d'événement : • Expiration • ExpirationNotification • NoChangeNotification Pour plus d'informations, consultez la section Configuration de EventBridge pour les stratégies de paramètres (p. 241).

Type d'événement :Fonctionnalité Exécuter la commande

Nom du type d'événement	Description des événements que vous pouvez ajouter à une règle
Notification de changement de statut d'appel de commande EC2	Le statut d'une commande envoyée à une instance gérée individuelle change.Vous pouvez ajouter une ou plusieurs des modifications de statut suivantes à une règle d'événement : • Success (Succès) • InProgress • TimedOut • Annulée • Failed (Échec)
Notification de changement de statut de commande EC2	Le statut global d'une commande change.Vous pouvez ajouter une ou plusieurs des modifications de statut suivantes à une règle d'événement : • Success (Succès) • InProgress • TimedOut • Annulée • Failed (Échec)

Référence : Expressions cron et rate pour Systems Manager

Lorsque vous créez une fenêtre de maintenance AWS Systems Manager ou une association State Manager, vous spécifiez la planification d'une exécution de la fenêtre ou de l'association. Vous pouvez spécifier une planification sous la forme d'une entrée temporelle, appelée expression cron ou une entrée basée sur la fréquence appelée expression de fréquence.

Lorsque vous créez une fenêtre de maintenance, vous pouvez spécifier un horodatage au format UTC (temps universel coordonné) afin qu'il s'exécute une fois à l'heure spécifiée. Les fenêtres de maintenance prennent également en charge les décalages de planification pour les expressions CRON uniquement. Un décalage de planification est le nombre de jours à attendre après la date et l'heure spécifiées par une expression CRON avant d'exécuter la fenêtre de maintenance. Par exemple, l'expression CRON/taux suivante planifie une fenêtre de maintenance pour une exécution le troisième mardi de chaque mois à 23h30.

```
cron(0 30 23 ? * TUE#3 *)
```

Si le décalage horaire est `2`, la fenêtre de maintenance ne s'exécutera qu'à 23h30 deux jours plus tard.

> **Note**
>
> Si vous créez une fenêtre de maintenance avec une expression cron qui cible un jour qui s'est déjà écoulé dans la période en cours, mais que vous ajoutez une date de décalage planifié qui

tombe dans le futur, la fenêtre de maintenance ne s'exécute pas dans la période. Il entrera en vigueur au cours de la période suivante. Par exemple, si vous spécifiez une expression cron qui aurait exécuté une fenêtre de maintenance hier et ajouté un décalage horaire de deux jours, la fenêtre de maintenance ne s'exécutera pas demain.

Lorsque vous créez une association ou une fenêtre de maintenance par programmation ou à l'aide d'un outil de ligne de commande tel que le AWS Command Line Interface (AWS CLI), vous devez spécifier un paramètre de planification avec une expression cron ou rate valide (ou un horodatage pour les fenêtres de maintenance) au format correct.

Lorsque vous utilisez la console AWS Systems Manager pour créer une fenêtre de maintenance ou une association, vous pouvez spécifier une planification à l'aide d'une expression cron ou rate valide. Vous pouvez également utiliser les outils de l'interface utilisateur qui simplifient le processus de création de votre calendrier.

Exemples de fenêtres de maintenance

Pour créer des fenêtres de maintenance à l'aide de l' AWS CLI, vous incluez le `--schedule` paramètre avec une expression cron ou rate ou un horodatage. Les exemples suivants utilisent le AWS CLI sur une machine Linux locale.

```
aws ssm create-maintenance-window \
    --name "My-Cron-Maintenance-Window" \
    --allow-unassociated-targets \
    --schedule "cron(0 16 ? * TUE *)" \
    --schedule-timezone "America/Los_Angeles" \
    --start-date 2021-01-01T00:00:00-08:00 \
    --end-date 2021-06-30T00:00:00-08:00 \
    --duration 4 \
    --cutoff 1
```

```
aws ssm create-maintenance-window \
    --name "My-Cron-Offset-Maintenance-Window" \
    --allow-unassociated-targets \
    --schedule "cron(0 30 23 ? * TUE#3 *)" \
    --duration 4 \
    --cutoff 1 \
    --schedule-offset 2
```

```
aws ssm create-maintenance-window \
    --name "My-Rate-Maintenance-Window" \
    --allow-unassociated-targets \
    --schedule "rate(7 days)" \
    --duration 4 \
    --schedule-timezone "America/Los_Angeles" \
    --cutoff 1
```

```
aws ssm create-maintenance-window \
    --name "My-TimeStamp-Maintenance-Window" \
    --allow-unassociated-targets \
    --schedule "at(2021-07-07T13:15:30)" \
    --duration 4 \
    --schedule-timezone "America/Los_Angeles" \
    --cutoff 1
```

Exemples d'associations

Pour créer des associations State Manager à l'aide de l'AWS CLI, vous incluez le paramètre `--schedule-expression` avec une expression cron ou rate, ou un horodatage. Les exemples suivants utilisent le AWS CLI sur une machine Linux locale.

```
aws ssm create-association \
    --association-name "My-Cron-Association" \
    --schedule-expression "cron(0 0 2 ? * SUN *)" \
    --targets Key=tag:ServerRole,Values=WebServer \
    --name AWS-UpdateSSMAgent
```

```
aws ssm create-association \
    --association-name "My-Rate-Association" \
    --schedule-expression "rate(7 days)" \
    --targets Key=tag:ServerRole,Values=WebServer \
    --name AWS-UpdateSSMAgent
```

```
aws ssm create-association \
    --association-name "My-Rate-Association" \
    --schedule-expression "at(2020-07-07T15:55:00)" \
    --targets Key=tag:ServerRole,Values=WebServer \
    --name AWS-UpdateSSMAgent \
    --apply-only-at-cron-interval
```

Note

Par défaut, lorsque vous créez une nouvelle association, le système l'exécute immédiatement après sa création, puis selon la planification que vous avez spécifiée. Préciser `--apply-only-at-cron-interval` afin que l'association ne s'exécute pas immédiatement après que vous l'avez créée.

Rubriques

- Informations générales sur les expressions cron et rate (p. 1394)
- Expressions cron et rate pour les associations (p. 1397)
- Expressions cron et rate pour les fenêtres de maintenance (p. 1398)

Informations générales sur les expressions cron et rate

Les expressions cron pour Systems Manager comportent six champs obligatoires. Un septième champ, le champ `Seconds` (le premier dans une expression cron), est facultatif. Ces champs sont séparés par un espace.

Exemples d'expressions cron

minutes	heures	Jour du mois	mois	Jour de la semaine	Année	Signification
0 USD	10	*	*	?	*	Exécuter à 10 h 00 (UTC) chaque jour
15	12	*	*	?	*	Exécuter à 12 h 15 (UTC) chaque jour
0 USD	18	?	*	MON-FRI	*	Exécuter à 18 h 00 (UTC) du

minutes	heures	Jour du mois	mois	Jour de la semaine	Année	Signification
						lundi au vendredi
0 USD	8	1	*	?	*	Exécuter à 8 h 00 (UTC) chaque 1er jour du mois
0/15	*	*	*	?	*	Exécuter toutes les 15 minutes
0/10	*	?	*	MON-FRI	*	Exécuter toutes les 10 minutes du lundi au vendredi
0/5	8-17	?	*	MON-FRI	*	Exécuter toutes les 5 minutes du lundi au vendredi entre 8 h 00 et 17 h 55 (UTC)

Valeurs prises en charge

Le tableau suivant montre les valeurs prises en charge pour les entrées cron obligatoires.

> **Note**
>
> Les expressions cron pour les associations ne prennent pas en charge toutes les ces valeurs. Pour plus d'informations, consultez la section Expressions cron et rate pour les associations (p. 1397).

Valeurs prises en charge pour les expressions cron

Field	Valeurs	Caractères génériques
minutes	0-59	, - * /
heures	0-23	, - * /
Jour du mois	1-31	, - * ? / L W
mois	1-12 ou JAN-DEC	, - * /
Day-of-week	1-7 ou SUN-SAT	, - * ? / L
Année	1970-2199	, - * /

> **Note**
>
> Vous ne pouvez pas spécifier de valeur dans les champs Jour-du-mois et Jour-de-la-semaine de la même expression cron. Si vous spécifiez une valeur dans l'un de ces champs, vous devez utiliser un signe ? (point d'interrogation) dans l'autre.

Wildcards

Le tableau suivant présente les valeurs de caractères génériques que les expressions cron prennent en charge.

Caractères génériques pris en charge pour les expressions cron

Caractère générique	Description
,	Le caractère générique , (virgule) inclut des valeurs supplémentaires. Dans le champ Mois, JAN,FEB,MAR englobe janvier, février et mars.
-	Le caractère générique - (tiret) spécifie des plages. Dans le champ Jour, 1-15 englobe les jours 1 à 15 du mois spécifié.
*	Le caractère générique * (astérisque) inclut toutes les valeurs du champ. Dans le champ Heures, * inclut toutes les heures.
/	Le caractère générique / (barre oblique) spécifie les incréments. Dans le champ Minutes, vous pouvez saisir 1/10 pour spécifier des intervalles de dix minutes, en partant de la première minute de l'heure. Donc 1/10 spécifie la première, la 11e, la 21e, la 31e minute, et ainsi de suite.
?	Le caractère générique ? (point d'interrogation) indique l'un ou l'autre. Dans le champ Jour-du-mois, vous pouvez entrer 7, et si peu importe pour vous le jour de la semaine auquel correspond le 7, vous pouvez entrer ? dans le champ Jour-de-la-semaine.
L	Le caractère générique L dans les champs Jour-du-mois ou Jour-de-la-semaine spécifie le dernier jour du mois ou de la semaine.
W	Le caractère générique W dans le champ Jour-du-mois spécifie un jour de la semaine. Dans le champ Jour-du-mois, 3W indique le jour le plus proche du troisième jour de la semaine du mois.

> **Note**
>
> Les expressions cron qui entraînent des fréquences d'une rapidité supérieure à cinq (5) minutes ne sont pas prises en charge. La prise en charge de la spécification simultanée d'une valeur de jour du mois et d'une valeur de jour de la semaine n'est pas complète. Pour le moment, vous devez utiliser un point d'interrogation « ? » dans l'un de ces champs.

Pour de plus amples informations sur les expressions cron, veuillez consulter CRON expression sur le site Web de Wikipedia.

Expressions de taux

Les expressions de fréquence comportent deux champs obligatoires. Les champs sont séparés par des espaces.

Champs obligatoires pour les expression rate

Field	Valeurs
Value	nombre positif, tel que `1` ou `15`

Unit	`minute`
	`minutes`
	`hour`
	`hours`
	`day`
	`days`

Si la valeur est égale à `1`, l'unité doit être au singulier. De même, pour les valeurs supérieures à `1`, l'unité doit être au pluriel. Par exemple, `rate(1 hours)` et `rate(5 hour)` ne sont pas valides, mais `rate(1 hour)` et `rate(5 hours)` sont valides.

Expressions cron et rate pour les associations

Cette section inclut des exemples d'expressions cron et rate pour les associations State Manager. Avant de créer l'une de ces expressions, soyez conscient des restrictions suivantes :

- Les associations prennent en charge uniquement les expressions cron suivantes : toutes les 1/2, 1, 2, 4, 8 ou 12 heures ; tous les jours ou toutes les semaines à une heure spécifique.
- Les associations prennent en charge uniquement les expressions rate suivantes : intervalles de 30 minutes ou plus et moins de 31 jours.
- Si vous spécifiez le champ facultatif `Seconds`, sa valeur ne peut être que de 0 (zéro). Par exemple: `cron(0 */30 * * * ? *)`

> **Note**
>
> Pour une association qui collecte des métadonnées pour Systems Manager Inventory, nous vous recommandons d'utiliser une expression rate.

Le tableau suivant présente des exemples d'expressions cron pour les associations avec les six champs obligatoires.

Exemples cron pour associations

Exemple	Details (Détails)
cron(0/30 * * * ? *)	Toutes les 30 minutes
cron(0 0/1 * * ? *)	Toutes les heures
cron(0 0/2 * * ? *)	Toutes les 2 heures

Exemple	Details (Détails)
cron(0 0/4 * * ? *)	Toutes les 4 heures
cron(0 0/8 * * ? *)	Toutes les 8 heures
cron(0 0/12 * * ? *)	Toutes les 12 heures
cron(15 13 ? * * *)	Tous les jours à 13 h 15
cron(15 13 ? * MON *)	Tous les lundis à 13 h 15

Voici quelques exemples d'expressions rate pour les associations.

Exemples rate pour les associations

Exemple	Details (Détails)
rate(30 minutes)	Toutes les 30 minutes
rate(1 hour)	Toutes les heures
rate(5 hours)	Toutes les 5 heures
rate(15 days)	Tous les 15 jours

Expressions cron et rate pour les fenêtres de maintenance

Cette section inclut des exemples d'expressions cron et rate pour les fenêtres de maintenance.

Contrairement aux associations State Manager, les fenêtres de maintenance prennent en charge toutes les expressions cron et rate. Cela inclut la prise en charge des valeurs dans le champ secondes. (Remarque : Zéro (0) n'est pas pris en charge dans le champ secondes).

Par exemple, l’expression cron à 6 champs exécute une fenêtre de maintenance à 9 h 30 chaque jour :

```
cron(30 09 ? * * *)
```

Si l'on ajoute une valeur au champ `Seconds`, l'expression cron suivante à 7 champs exécute une une fenêtre de maintenance tous les jours à 9 h 30 et 24 s :

```
cron(24 30 09 ? * * *)
```

Le tableau suivant fournit des exemples supplémentaires d'expressions cron à 6 champs pour les fenêtres de maintenance.

Exemples cron pour les fenêtres de maintenance

Exemple	Details (Détails)
cron(0 2 ? * THU#3 *)	2 h 00 le troisième jeudi de chaque mois
cron(15 10 ? * * *)	10 h 15 tous les jours

Exemple	Details (Détails)
cron(15 10 ? * MON-FRI *)	10 h 15 chaque lundi, mardi, mercredi, jeudi et vendredi
cron(0 2 L * ? *)	2 h 00 le dernier jour de chaque mois
cron(15 10 ? * 6L *)	10 h 15 le dernier vendredi de chaque mois

Le tableau suivant fournit des exemples rate pour les fenêtres de maintenance.

Exemples rate pour les fenêtres de maintenance

Exemple	Details (Détails)
rate(30 minutes)	Toutes les 30 minutes
rate(1 hour)	Toutes les heures
rate(5 hours)	Toutes les 5 heures
rate(25 days)	Tous les 25 jours

Référence : ec2messages, ssmmessages et autres appels d'API

Si vous surveillez les appels d'API, il se peut que vous constatiez des appels vers les API suivantes.

- ec2messages:AcknowledgeMessage
- ec2messages:DeleteMessage
- ec2messages:FailMessage
- ec2messages:GetEndpoint
- ec2messages:GetMessages
- ec2messages:SendReply
- ssmmessages:CreateControlChannel
- ssmmessages:CreateDataChannel
- ssmmessages:OpenControlChannel
- ssmmessages:OpenDataChannel
- ssm:UpdateInstanceInformation
- ssm:ListInstanceAssociations
- ssm:DescribeInstanceProperties
- ssm:DescribeDocumentParameters

Ces appels spéciaux sont utilisés par Systems Manager pour diverses opérations.

Appels vers les API ec2messages

Appels vers `ec2messages:*` Les API sont des appels à la Amazon Message Delivery Service point de terminaison. Systems Manager utilise ce point de terminaison pour passer des appels depuis Agent SSM à la Systems Manager dans le cloud. Ce point de terminaison est requis pour envoyer et recevoir des

commandes. Pour plus d'informations, consultez Actions, ressources et clés de condition pour Amazon Message Delivery Service.

Appels vers les API ssmmessages

Systems Manager utilise le point de terminaison `ssmmessages` pour effectuer des appels depuis l'Agent SSM vers le service Session Manager dans le cloud. Ce point de terminaison est requis pour créer et supprimer des canaux de session avec le service Session Manager dans le cloud.

Appels d'API liés aux instances

`UpdateInstanceInformation` : l'Agent SSM appelle le service Systems Manager dans le cloud toutes les cinq minutes pour fournir des informations de pulsation. Cet appel est nécessaire pour maintenir une pulsation avec l'agent afin que le service sache que l'agent fonctionne comme prévu.

`ListInstanceAssociations`: L'agent appelle cette API pour voir si une nouvelle Systems Manager State Manager est disponible. Cette API est requise pour que State Manager fonctionne.

`DescribeInstanceProperties` et `DescribeDocumentParameters` : Systems Manager appelle ces API pour rendre des nœuds spécifiques dans la console Amazon EC2. L'API `DescribeInstanceProperties` affiche le nœud Instances gérées dans le volet de navigation. L'API `DescribeDocumentParameters` affiche le nœud Documents dans le volet de navigation.

Référence : Création de chaînes de date et d'heure formatées pour Systems Manager

De nombreuses actions d'API AWS Systems Manager acceptent des filtres pour limiter le nombre de résultats renvoyés par une demande. Certaines de ces actions d'API acceptent des filtres qui nécessitent une chaîne formatée pour représenter une date et une heure spécifiques. Par exemple, l'action d'API `DescribeSessions` accepte les clés `InvokedAfter` et `InvokedBefore` parmi les valeurs valides pour un objet `SessionFilter`. L'action d'API `DescribeAutomationExecutions` constitue un autre exemple, qui accepte les clés `StartTimeBefore` et `StartTimeAfter` parmi les valeurs valides pour un objet `AutomationExecutionFilter`. Les valeurs que vous fournissez pour ces clés lorsque vous filtrez vos demandes doivent correspondre à la norme ISO 8601. Pour de plus amples informations sur la norme ISO 8601, veuillez consulter ISO 8601.

Ces chaînes de date et d'heure formatées ne sont pas limitées aux filtres. Il existe également des actions d'API qui nécessitent une chaîne formatée ISO 8601 pour représenter une date et une heure spécifiques lorsqu'une valeur est fournie pour un paramètre de demande. Par exemple, le paramètre de demande `AtTime` pour l'action `GetCalendarState`. Ces chaînes sont difficiles à créer. Utilisez les exemples de cette rubrique pour créer des chaînes de date et d'heure formatées à utiliser avec les actions d'API Systems Manager.

Mise en forme de chaînes de date et d'heure pour Systems Manager

Voici un exemple de chaîne de date et d'heure formatée conformément à la norme ISO 8601.

```
2020-05-08T15:16:43Z
```

L'exemple représente le 8 mai 2020 à 15 h 16, heure UTC. La partie date calendaire de la chaîne est représentée par une année à quatre chiffres, un mois à deux chiffres et un jour à deux chiffres séparés par des tirets. Voici le format.

```
YYYY-MM-DD
```

La partie heure de la chaîne est délimitée par la lettre « T » (pour Time). Elle est représentée par la valeur des heures sur deux chiffres, la valeur des minutes sur deux chiffres et la valeur des secondes sur deux chiffres, séparées par deux points. Voici le format.

```
hh:mm:ss
```

La partie heure de la chaîne se termine par la lettre « Z », indiquant la norme UTC.

Création de chaînes de date et d'heure personnalisées pour Systems Manager

Vous pouvez créer des chaînes de date et d'heure personnalisées à partir de votre machine locale à l'aide de votre outil de ligne de commande préféré. La syntaxe que vous utilisez pour créer une chaîne de date et d'heure au format ISO 8601 diffère selon le système d'exploitation de votre machine locale. Voici des exemples de la façon dont vous pouvez utiliser `date` avec les utilitaires principaux de GNU sous Linux, ou PowerShell sous Windows pour créer une chaîne de date et d'heure au format ISO 8601.

coreutils

```
date '+%Y-%m-%dT%H:%M:%SZ'
```

PowerShell

```
(Get-Date).ToString("yyyy-MM-ddTH:mm:ssZ")
```

Lorsque vous utilisez des actions d'API Systems Manager, vous devrez peut-être créer des chaînes de date et d'heure historiques à des fins de reporting ou de dépannage. Voici des exemples de création et d'utilisation de chaînes de date et d'heure au format ISO 8601 historiques personnalisées pour Outils AWS pour PowerShell et l'AWS CLI.

AWS CLI

- Récupérer la dernière semaine de l'historique des commandes d'un document SSM.

```
lastWeekStamp=$(date '+%Y-%m-%dT%H:%M:%SZ' -d '7 days ago')

docFilter='{"key":"DocumentName","value":"AWS-RunPatchBaseline"}'
timeFilter='{"key":"InvokedAfter","value":'\"$lastWeekStamp\"'}'

commandFilters=[$docFilter,$timeFilter]

aws ssm list-commands \
    --filters $commandFilters
```

- Récupérer la dernière semaine de l'historique d'exécution de l'automatisation.

```
lastWeekStamp=$(date '+%Y-%m-%dT%H:%M:%SZ' -d '7 days ago')

aws ssm describe-automation-executions \
    --filters Key=StartTimeAfter,Values=$lastWeekStamp
```

- Récupérer le dernier mois de l'historique de session.

```
lastWeekStamp=$(date '+%Y-%m-%dT%H:%M:%SZ' -d '30 days ago')

aws ssm describe-sessions \
    --state History \
    --filters key=InvokedAfter,value=$lastWeekStamp
```

Outils AWS pour PowerShell

- Récupérer la dernière semaine de l'historique des commandes d'un document SSM.

```
$lastWeekStamp = (Get-Date).AddDays(-7).ToString("yyyy-MM-ddTH:mm:ssZ")

$docFilter = @{
    Key="DocumentName"
    Value="AWS-InstallWindowsUpdates"
}
$timeFilter = @{
    Key="InvokedAfter"
    Value=$lastWeekStamp
}

$commandFilters = $docFilter,$timeFilter

Get-SSMCommand `
    -Filters $commandFilters
```

- Récupérer la dernière semaine de l'historique d'exécution de l'automatisation.

```
$lastWeekStamp = (Get-Date).AddDays(-7).ToString("yyyy-MM-ddTH:mm:ssZ")

Get-SSMAutomationExecutionList `
    -Filters @{Key="StartTimeAfter";Values=$lastWeekStamp}
```

- Récupérer le dernier mois de l'historique de session.

```
$lastWeekStamp = (Get-Date).AddDays(-30).ToString("yyyy-MM-ddTH:mm:ssZ")

Get-SSMSession `
    -State History `
    -Filters @{Key="InvokedAfter";Value=$lastWeekStamp}
```

Cas d'utilisation et bonnes pratiques

Cette rubrique répertorie les cas d'utilisation courants et les bonnes pratiques pour les fonctionnalités AWS Systems Manager. Si elle est disponible, cette rubrique comporte aussi des liens vers de la documentation technique et des billets de blogs pertinents.

> Note
>
> Le titre de chaque section ici est un lien actif vers la section correspondante dans la documentation technique.

Automation (p. 304)

- Créez des runbooks en libre-service pour l'infrastructure sous forme de documents d'Automation.
- Utilisez l'automatisation pour simplifier la création AMIs à partir d'AWS Marketplace ou personnalisée AMIs à l'aide de SSM documents publics ou en créant vos propres flux de travail.
- Créez et gérez à AMIs (p. 632) l'aide `AWS-UpdateLinuxAmi` de l' et des documents `AWS-UpdateWindowsAmi` Automation, ou à l'aide des documents Automation personnalisés que vous créez.

Inventory (p. 791)

- Utilisez Systems Manager Inventory avec AWS Config pour contrôler les configurations des applications dans le temps.

Fenêtres de maintenance (p. 688)

- Définissez une planification pour exécuter des actions potentiellement perturbatrices, comme le correctif du système d'exploitation, les mises à jour du pilote ou les installations de logiciels.
- Pour plus d'informations sur les différences entre State Manager et Fenêtres de maintenance , consultez Choisir entre State Manager etFenêtres de maintenance (p. 1405) .

Parameter Store (p. 218)

- Utilisez Parameter Store pour gérer de façon centralisée les paramètres de configuration généraux.
- Utilisez Parameter Store pour chiffrer et gérer des codes secrets en utilisant AWS Key Management Service.
- Référencez des secrets AWS Secrets Manager à partir des paramètres Parameter Store (p. 134).
- Utilisez Parameter Store avec les définitions de tâches ECS pour stocker des codes secrets.

Gestionnaire de correctifs (p. 1047)

- Utilisez Gestionnaire de correctifs pour déployer des correctifs à grande échelle et augmenter la visibilité de la conformité du parc sur vos instances.
- Intégrez Gestionnaire de correctifs à AWS Security Hub (p. 1145) pour recevoir des alertes lorsque les instances de votre parc ne sont plus conformes et surveiller l'état des correctifs de vos parcs du point de vue de la sécurité. Des frais sont facturés pour l'utilisation d' Security Hub . Pour de plus amples informations, veuillez consulter Tarification .

Fonctionnalité Exécuter la commande (p. 941)

- Gérer les instances à grande échelle sans accès SSH en utilisant la fonctionnalité Exécuter la commande d'EC2.

- Contrôlez tous les appels d'API effectués par ou au nom de la fonctionnalité Fonctionnalité Exécuter la commande à l'aide d'AWS CloudTrail.
- Lorsque vous exécutez une commande dans Fonctionnalité Exécuter la commande, n'incluez aucune information sensible formatée en texte brut, comme les mots de passe, les données de configuration ou d'autres secrets. L'ensemble de l'activité de l'API Systems Manager de votre compte est enregistrée dans un compartiment Amazon S3, dans des journaux AWS CloudTrail. Cela signifie que tout utilisateur ayant accès à ce compartiment S3 peut afficher les valeurs en texte brut de ces secrets. Pour cette raison, nous vous recommandons fortement de créer et d'utiliser des paramètres `SecureString` pour chiffrer les données sensibles que vous utilisez dans vos opérations Systems Manager.

  Pour plus d'informations, consultez la section Restriction de l'accès aux paramètres Systems Manager à l'aide des stratégies IAM (p. 223).

  > Note
  >
  > Par défaut, les fichiers journaux livrés par CloudTrail à votre compartiment sont chiffrés par Amazon côté serveur avec des clés de chiffrement gérées par Amazon S3 (SSE-S3). Pour fournir une couche de sécurité qui soit directement gérable, vous pouvez utiliser à la place le chiffrement côté serveur avec des clés gérées par AWS KMS (SSE-KMS) pour vos fichiers journaux CloudTrail.
  > Pour plus d'informations, consultez Chiffrement des fichiers CloudTrail journaux avec des clés gérées par AWS KMS (SSE-KMS) dans le AWS CloudTrail User Guide .
- Utilisez les cibles et les fonctions de contrôle du débit dans Fonctionnalité Exécuter la commande pour effectuer une exécution des commandes par étapes (p. 955).
- Utilisez des autorisations d'accès plus détaillées pour la fonctionnalité Fonctionnalité Exécuter la commande (et l'ensemble des fonctionnalités Systems Manager) à l'aide de stratégies AWS Identity and Access Management (IAM) (p. 1307).

State Manager (p. 987)

- Mettez à jour l'Agent SSM au moins une fois par mois à l'aide du document préconfiguré AWS-UpdateSSMAgent (p. 1043).
- Amorcez les instances EC2 au lancement avec EC2Config pour Windows.
- (Windows) Chargez le module PowerShell ou DSC sur et utilisez AWS-InstallPowerShellModule.Amazon S3
- Utilisez les balises Amazon EC2 pour créer des groupes d'applications pour vos instances. Puis, ciblez les instances à l'aide du `Targets` paramètre au lieu de spécifier une instance individuelleIDs.
- Corrigez automatiquement les résultats générés par Amazon Inspector en utilisant Systems Manager.
- Utilisez un référentiel de configuration centralisé pour l'ensemble de vos documents SSM, et partagez les documents dans l'ensemble de votre organisation (p. 1275).
- Pour plus d'informations sur les différences entre State Manager et Fenêtres de maintenance , consultez Choisir entre State Manager etFenêtres de maintenance (p. 1405) .

Instances gérées (p. 851)

- Systems Manager nécessite des références horaires précises pour pouvoir effectuer ses opérations. Si la date et l'heure de votre instance ne sont pas correctement définies, il est possible que la date de la signature ne corresponde pas à vos demandes d'API. Dans certains cas, cela entraînera des erreurs ou des fonctionnalités incomplètes. Par exemple, les instances ayant des paramètres de temps incorrects ne seront pas inclus dans vos listes d'instances gérées.

  Pour plus d'informations sur le paramétrage du temps sur vos instances, consultez l'une des rubriques suivantes :

  - Réglage de l'heure pour votre instance Linux

- Réglage de l'heure pour une instance Windows

Choisir entre State Manager etFenêtres de maintenance

State Manager et Fenêtres de maintenance peuvent effectuer des types similaires de mise à jour sur vos instances gérées. Votre choix dépend de la nécessité d'automatiser la conformité du système ou d'effectuer des tâches prioritaires et urgentes uniquement pendant les périodes que vous spécifiez.

State Manageret Fenêtres de maintenance : Cas d'utilisation des clés

State Managerdéfinit et gère la configuration d'état souhaitée pour les instances et AWS les ressources gérées au sein de votre compte. Vous pouvez définir des combinaisons de configurations et de cibles en tant qu'objets d'association. State Manager est la capacité recommandée si vous souhaitez conserver toutes les instances gérées de votre compte dans un état cohérent, utiliser Amazon EC2 Auto Scaling (Auto Scaling) pour générer de nouvelles instances ou avoir des exigences strictes en matière de rapports de conformité pour les instances gérées de votre compte.

Les principaux cas d'utilisation pour State Manager sont les suivants :

- Auto ScalingScénarios : State Manager peut surveiller toutes les nouvelles instances lancées dans un compte manuellement ou via Auto Scaling des groupes. En cas d'associations dans le compte ciblant cette nouvelle instance (via des balises ou toutes les instances), cette association particulière est automatiquement appliquée à la nouvelle instance.
- Rapports de conformité : State Manager peut influencer les rapports de conformité des états souhaités pour les ressources de votre compte.
- Prise en charge de toutes les instances : State Manager peut cibler toutes les instances d'un compte donné.

Une fenêtre de maintenance effectue une ou plusieurs actions sur AWS les ressources au sein d'une fenêtre temporelle donnée. Vous pouvez définir une seule fenêtre de maintenance avec des heures de début et de fin. Vous pouvez spécifier plusieurs tâches à exécuter dans cette fenêtre de maintenance. Fenêtres de maintenance est la capacité recommandée si vos opérations de priorité élevée incluent l'application de correctifs à vos instances gérées, l'exécution de plusieurs types de tâches sur vos instances pendant une période de mise à jour ou le contrôle du moment où les opérations de mise à jour peuvent être exécutées sur vos instances.

Les principaux cas d'utilisation pour Fenêtres de maintenance sont les suivants :

- Exécution de plusieurs documents : Les fenêtres de maintenance peuvent exécuter plusieurs tâches. Chaque tâche peut utiliser un type de document différent. Par conséquent, vous pouvez créer des flux de travail complexes à l'aide de différentes tâches au sein d'une même fenêtre de maintenance.
- Application de correctifs : Une fenêtre de maintenance peut fournir la prise en charge de l'application de correctifs pour toutes les instances gérées d'un compte balisé avec une balise ou un groupe de ressources spécifique. Étant donné que l'application de correctifs implique généralement de réduire les instances (par exemple, de supprimer les instances d'un équilibreur de charge), d'appliquer des correctifs et de procéder au post-traitement (remettre des instances en production), l'application des correctifs peut être réalisée sous la forme d'une série de tâches dans une fenêtre de temps de correctif donnée.
- Actions de fenêtrage : Les fenêtres de maintenance peuvent démarrer un ou plusieurs ensembles d'actions dans une fenêtre de temps spécifique. Les fenêtres de maintenance ne démarrent pas en

dehors de cette fenêtre. Les actions déjà démarrées se poursuivront jusqu'à ce qu'elles soient terminées, même si elles se terminent en dehors de la fenêtre horaire.

Le tableau suivant compare les principales fonctions d' State Manager et Fenêtres de maintenance d' .

Fonctionnalité	State Manager	Fenêtres de maintenance
AWS CloudFormation intégration	AWS CloudFormationLes modèles prennent en charge State Manager les associations.	AWS CloudFormationLes modèles prennent en charge les fenêtres de maintenance, les cibles de fenêtre et les tâches de fenêtre.
Conformité	Chaque State Manager association signale la conformité par rapport à l'état souhaité de la ressource ciblée. Vous pouvez utiliser le tableau de bord de conformité pour regrouper et afficher la conformité signalée.	Ne s'applique pas.
Intégration de la gestion de configuration	State Managerprend en charge les solutions d'état souhaitées externes telles que Microsoft PowerShell Desired State Configuration (DSC), les manuels stratégiques Ansible et les recettes Chef. Vous pouvez utiliser State Manager des associations pour vérifier que les solutions de gestion de configuration fonctionnent et appliquer leurs modifications de configuration à vos instances lorsque vous êtes prêt.	Ne s'applique pas.
Documents	State ManagerLes configurations peuvent être définies en tant que documents de stratégie (pour la collecte d'informations d'inventaire), documents d'automatisation (pour AWS des ressources telles que des Amazon Simple Storage Service compartiments) ou documents de commande (pour des instances gérées).	Fenêtres de maintenanceLes configurations peuvent être définies comme des documents d'automatisation (actions en plusieurs étapes avec des flux de travail d'approbation facultatifs) ou des documents de commande (état souhaité pour les instances gérées).
Surveillance	State Managersurveille les modifications apportées à la configuration, à l'association ou à l'état d'une instance (par exemple, de nouvelles instances en ligne). Quand State Manager détecte ces modifications, l'association donnée est réappliquée aux	Ne s'applique pas.

Fonctionnalité	State Manager	Fenêtres de maintenance
	instances initialement ciblées avec cette association.	
Priorités au sein des tâches	Ne s'applique pas.	Les tâches au sein d'une fenêtre de maintenance peuvent se voir attribuer une priorité. Toutes les tâches avec la même priorité sont exécutées en parallèle. Les tâches avec des priorités inférieures sont exécutées uniquement une fois que les tâches avec des priorités supérieures ont atteint un état final. Il n'existe aucun moyen d'exécuter des tâches de manière conditionnelle. Une fois qu'une tâche de priorité supérieure atteint son état final, la tâche de priorité suivante s'exécute, quel que soit l'état de la tâche précédente.
Contrôles de sécurité	State Managerprend en charge deux contrôles de sécurité lors du déploiement de configurations dans un grand parc. Vous pouvez utiliser la simultanéité maximale pour définir le nombre d'instances ou de ressources simultanées auquel la configuration doit être appliquée. Vous pouvez définir un taux d'erreurs maximum qui peut être utilisé pour suspendre l'State Managerassociation si un certain nombre ou pourcentage d'erreurs se produit dans l'ensemble du parc.	Les fenêtres de maintenance prennent en charge deux contrôles de sécurité lors du déploiement de configurations dans un parc d'instances volumineux. Vous pouvez utiliser la simultanéité maximale pour définir le nombre d'instances ou de ressources simultanées auquel la configuration doit être appliquée. Vous pouvez définir un taux d'erreur maximal qui peut être utilisé pour suspendre les actions dans une fenêtre de maintenance si un certain nombre ou pourcentage d'erreurs se produit dans l'ensemble du parc.

Fonctionnalité	State Manager	Fenêtres de maintenance
Planification	Vous pouvez exécuter State Manager des associations à la demande, à un intervalle cron donné, à un taux donné ou une fois lorsqu'elles sont créées. Cela est utile si vous souhaitez conserver l'état souhaité de vos ressources de manière cohérente et en temps opportun.	Les fenêtres de maintenance prennent en charge de nombreuses options de planification, y compris `at` les expressions (par exemple, `"at(2021-07-07T13:15:30)"`), les expressions cron et rate, cron avec décalages, les heures de début et de fin du moment où les fenêtres de maintenance doivent être exécutées, et les heures limites pour spécifier quand arrêter la planification dans une fenêtre temporelle donnée.
Ciblage	State ManagerLes associations peuvent cibler une ou plusieurs instances à l'aide d'un ID d'instance, d'une balise ou dState Manager'un groupe de ressources. peut cibler toutes les instances gérées au sein d'un compte donné.	Les fenêtres de maintenance peuvent cibler une ou plusieurs instances à l'aide d'instanceIDss , de balises ou de groupes de ressources.
Tâches au sein des fenêtres de maintenance	Ne s'applique pas.	Les fenêtres de maintenance peuvent prendre en charge une ou plusieurs tâches dans lesquelles chaque tâche cible une action spécifique d'automatisation ou de document de commande. Toutes les tâches au sein d'une fenêtre de maintenance s'exécutent en parallèle, sauf si des priorités différentes sont définies pour des tâches différentes. Dans l'ensemble, les fenêtres de maintenance prennent en charge quatre types de tâches : • Commandes de Fonctionnalité Exécuter la commande Systems Manager • Flux de travail d'automatisation Systems Manager • Fonctions AWS Lambda • Tâches AWS Step Functions

Historique du document

Le tableau suivant décrit les modifications importantes apportées à la documentation depuis la dernière version d'AWS Systems Manager. Pour recevoir les notifications concernant les mises à jour de cette documentation, abonnez-vous à un flux RSS.

> **Important**
>
> Une version mise à jour de l'Agent SSM est publiée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Si une version antérieure de l'agent est en cours d'exécution sur une instance, certains processus de l'Agent SSM peuvent échouer. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir l'Agent SSM à jour sur vos instances. Pour de plus amples informations, veuillez consulter Automatisation des mises à jour versAgent SSM (p. 109). Pour être alerté des mises à jour de Agent SSM, abonnez-vous à la page Agent SSM Release Notes sur GitHub.

- Version d'API : 2014-11-06

update-history-change	update-history-description	update-history-date
Prise en charge de Ubuntu Server 20.10 (p. 1409)	prend AWS Systems Manager désormais en charge la version à court terme Ubuntu Server 20.10. Pour de plus amples informations, consultez les rubriques suivantes : • Systèmes d'exploitation pris en charge • Installer Agent SSM pour un environnement hybride (Linux) • Installation manuelle de Agent SSM sur les instances Ubuntu Server • Vérification de Agent SSM l'état et démarrage de l'agent	October 22, 2020
Nouvelle rubrique : Activer les profils shell configurables (p. 1409)	Vous pouvez désormais activer des profils shell configurables avec Session Manager. En activant des profils shell configurables, vous pouvez personnaliser des préférences au sein de sessions telles que des préférences shell, des variables d'environnement, des répertoires de travail et l'exécution de plusieurs commandes lorsqu'une session est démarrée. Pour plus d'informations, consultez Activer les profils shell configurables.	October 21, 2020

Les résultats de conformité des correctifs indiquent désormais qui CVEs sont résolus par quels correctifs (p. 1409)	Pour la plupart des systèmes Linux pris en charge, lorsque vous affichez les résultats de conformité des correctifs pour vos instances gérées, les détails que vous pouvez maintenant afficher signalent les problèmes de bulletin Vulnérabilités et exposition courantes (CVE) qui sont résolus par les correctifs disponibles. Ces informations peuvent vous aider à déterminer l'urgence à laquelle vous avez besoin d'installer un correctif manquant ou ayant échoué. Pour plus d'informations, consultez Affichage des résultats de conformité des correctifs.	October 20, 2020
Prise en charge étendue des métadonnées de correctifs Linux (p. 1409)	Vous pouvez désormais afficher de nombreux détails sur les correctifs Linux disponibles dans Gestionnaire de correctifs . Vous pouvez choisir d'afficher les données de correctif telles que l'architecture, l'époque, la version, l'ID CVE, l'ID Advisory, l'ID Bugzilla, le référentiel, etc. En outre, l'action DescribeAvailablePatches d'API a été mise à jour pour prendre en charge les systèmes d'exploitation Linux et le filtrage en fonction de ces types de métadonnées de correctifs nouvellement disponibles. Pour de plus amples informations, consultez les rubriques suivantes : • Affichage des correctifs disponibles • DescribeAvailablePatcheset correctif dans leRéférence d'API AWS Systems Manager • describe-available-patches dans le AWS Systems Manager section of the AWS CLI Command Reference.	October 16, 2020

Session Managerpour la AWS CLI version 1.2.7.0 (p. 1409)	Une nouvelle version du Session Manager plug-in pour l' AWS CLI a été publiée. Pour plus d'informations, consultez la dernière version et l'historique des versions du Session Manager plug-in.	October 15, 2020
Nouvelle rubrique : Schéma du document de session (p. 1409)	Le nouveau schéma de document de session décrit les éléments de schéma d'un document de session. Ces informations peuvent vous aider à créer des documents de session personnalisés dans lesquels vous spécifiez des préférences pour les types de sessions que vous utilisez avec Session Manager .	October 15, 2020
Nouvelle rubrique : Recherche en texte libre pour les documents SSM (p. 1409)	La zone de recherche sur la Systems Manager page Documents prend désormais en charge la recherche en texte libre. La recherche en texte libre compare le terme recherché ou les termes que vous saisissez avec le nom du document dans chaque SSM document. Pour plus d'informations, consultez Utilisation de la recherche en texte libre.	October 15, 2020
Nouvelle rubrique : Dépannage de la disponibilité Amazon EC2 des instances gérées (p. 1409)	La nouvelle rubrique Dépannage de la disponibilité Amazon EC2 des instances gérées vous aide à déterminer pourquoi une Amazon EC2 instance que vous avez confirmée est en cours d'exécution n'est pas disponible dans les listes des instances gérées disponibles dans Systems Manager .	October 6, 2020

Réorganisation du chapitre sur Parameter Store (p. 1409)	Pour vous aider à trouver plus efficacement les informations dont vous avez besoin, nous avons réorganisé le contenu du Parameter Store chapitre du Guide de l'utilisateur AWS Systems Manager . La plupart du contenu est désormais organisé dans les sections Configuration Parameter Store et utilisation d' Parameter Store . En outre, la rubrique AWS Systems Manager Parameter Store a été étendue pour inclure les sections suivantes : • Comment mon organisation peut-elle tirer parti de Parameter Store ? • À qui est destiné Parameter Store ? • Quelles sont les fonctions d'Parameter Store ? • Qu'est-ce qu'un paramètre ?	October 1, 2020
Nouvelles rubriques liées à la conformité des correctifs (p. 1409)	Les rubriques suivantes ont été ajoutées pour vous aider à identifier les instances gérées qui ne sont pas conformes aux correctifs, à comprendre les différents types d'analyses de conformité des correctifs et à prendre les mesures appropriées pour rendre vos instances conformes. • Identification des instances hors conformité • Application de correctifs aux instances non conformes • Affichage des résultats de conformité des correctifs	September 24, 2020
Agent SSM version 3.0 (p. 1409)	Systems Managera lancé une nouvelle version d' Agent SSM . Pour plus d'informations, consultez Agent SSM la version 3.	September 21, 2020

Rubriques nouvelles et mises à jour : Amazon EventBridge remplace CloudWatch Events la gestion des événements (p. 1409)	CloudWatch Eventset EventBridge sont le même service sous-jacent et la même API, mais EventBridge offrent plus de fonctionnalités et constitue désormais le meilleur moyen de gérer vos événements dans AWS . (Les modifications que vous apportez dans l'un CloudWatch ou l'autre ou EventBridge apparaîtront dans chaque console.) Les références à CloudWatch Events et aux procédures existantes de l' Guide de l'utilisateur AWS Systems Manager ont été mises à jour pour refléter la EventBridge prise en charge d' . En outre, les nouvelles rubriques suivantes ont été ajoutées. • Surveillance des Systems Manager événements • Configuration EventBridge d' pour les Systems Manager événements • Systems Manager Exemples de type cible • Référence : modèles et types Amazon EventBridge d'événements pour Systems Manager	September 18, 2020
OpsItemarchivage (p. 1409)	archive OpsCenter automatiquement OpsItems après 36 mois, quel que soit le statut.	September 17, 2020
Intégration d' AWS Security Hub et d'Gestionnaire de correctifs (p. 1409)	Vous pouvez désormais intégrer Gestionnaire de correctifs à AWS Security Hub . fournit une vue complète de votre état de sécurité dans Security Hub et vous aide à vérifier votre environnement par rapport aux normes et aux bonnes pratiques du secteur de la sécurité.AWS Une fois intégré à Gestionnaire de correctifs , Security Hub surveille l'état d'application des correctifs de vos flottes du point de vue de la sécurité. Pour plus d'informations, consultez Intégration à Gestionnaire de correctifs AWS Security Hub .	September 17, 2020

Pseudo-paramètres de la fenêtre de maintenance : Nouveaux types de ressources pris en charge pour {{TARGET_ID}} et{{RESOURCE_ID}} (p. 1409)	Lorsque vous enregistrez une tâche de fenêtre de maintenance, vous utilisez l'`--task-invocation-parameters`option pour spécifier les paramètres qui sont spécifiques à chacun des quatre types de tâche. Vous pouvez également référencer certaines valeurs à l'aide de la syntaxe des pseudo-paramètres, comme `{{TARGET_ID}}` et `{{RESOURCE_ID}}` . Quand la tâche de fenêtre de maintenance s'exécute, elle transmet les valeurs correctes au lieu des espaces réservés des pseudo-paramètres. Deux types de ressources supplémentaires sont désormais disponibles pour une utilisation avec les pseudo-paramètres `{{TARGET_ID}}` et `{{RESOURCE_ID}}` . Vous pouvez désormais utiliser les types de ressources `AWS::RDS::DBInstance` et `AWS::SSM::ManagedInstance` ces deux pseudo-paramètres. Pour plus d'informations sur les pseudo-paramètres de la fenêtre de maintenance, consultez À propos des pseudo-paramètres.	September 14, 2020

Corriger les instances à la demande avec la nouvelle option « Patch now (Corriger maintenant) » (p. 1409)	Vous pouvez désormais utiliser la Systems Manager console pour corriger les instances ou rechercher les correctifs manquants à tout moment. Vous pouvez le faire sans avoir à créer ou à modifier une planification, ou à spécifier des options complètes de configuration d'application de correctifs pour répondre à un besoin immédiat d'application de correctifs. Il vous suffit de spécifier si les correctifs doivent être analysés ou installés, et d'identifier les instances cibles pour l'opération. applique Gestionnaire de correctifs automatiquement la référence de correctifs par défaut actuelle pour vos types d'instance et applique les options de bonnes pratiques pour le nombre d'instances corrigées en une seule fois, ainsi que le nombre d'erreurs autorisées avant que l'opération échoue. Pour plus d'informations, consultez Correction d'instances à la demande.	September 9, 2020
Nouvelle rubrique : Vérification de Agent SSM l'état et démarrage de l'agent (p. 1409)	La nouvelle rubrique Vérification du Agent SSM statut et démarrage de l'agent fournit des commandes pour vérifier si Agent SSM est en cours d'exécution sur chaque système d'exploitation pris en charge. Elle fournit également les commandes pour démarrer l'agent s'il n'est pas en cours d'exécution.	September 7, 2020
prend Gestionnaire de correctifs désormais en charge Ubuntu Server 20.04 LTS (p. 1409)	Vous pouvez désormais utiliser Gestionnaire de correctifs pour corriger les instances Ubuntu Server 20.04 LTS. Pour de plus amples informations, consultez les rubriques suivantes : • Sélection des correctifs de sécurité (p. 1049) • Installation des correctifs (p. 1056) • Fonctionnement des règles de référence de correctif sur Ubuntu Server (p. 1074)	August 31, 2020

Nouvelle rubrique pour les cas d'utilisation et les bonnes pratiques (p. 1409)	Nous avons ajouté une nouvelle rubrique pour aider les utilisateurs à comprendre rapidement les différences entre Fenêtres de maintenance et State Manager . Pour plus d'informations, consultez Choisir entre State Manager et Fenêtres de maintenance .	August 28, 2020
Nouvelles fonctions OpsCenter (p. 1409)	OpsCentercomprend de nouvelles fonctionnalités pour vous aider à localiser et à exécuter rapidement les documents Automation pour corriger les problèmes. Pour plus d'informations, consultez Fonctions du document Automation dans OpsCenter .	August 19, 2020
Nouvelle source de données dans Explorer : Dossiers AWS Support (p. 1409)	affiche Explorer désormais des informations sur les demandes AWS Support. Vous devez disposer d'un compte d'entreprise ou d'entreprise configuré avec AWS Support. Pour plus d'informations, consultez Modification des sources de données Systems Manager Explorer.	August 13, 2020
fournit Distributor désormais un package tiers de Trend Micro. (p. 1409)	inclut Distributor désormais un package tiers de Trend Micro. Vous pouvez utiliser Distributor pour installer l'agent Trend Micro Cloud One sur vos instances gérées. Trend Micro Cloud One vous aide à sécuriser vos charges de travail dans le cloud. Pour plus d'informations, consultez AWS Systems Manager Distributor.	August 12, 2020
Le plug-in de document aws:configurePackage inclut désormais le additionalArguments paramètre . (p. 1409)	Le plug-in du document Systems Manager de commande prend `aws:configurePackage` désormais en charge l'ajout de paramètres supplémentaires à vos scripts (installation, désinstallation et mise à jour) avec le nouveau `additionalArguments` paramètre. Pour plus d'informations, consultez la rubrique aws:configurePackage.	August 11, 2020

AppConfigcontenu déplacé dans un guide de l'utilisateur distinct (p. 1409)	Les informations sur AWS AppConfig ont été déplacées dans un guide de l'utilisateur distinct. Pour plus d'informations, consultez Qu'est-ce quAppConfig'AWS ? dispose AppConfig également d'une page de destination de documentation distincte AppConfig avec des liens vers le guide de l'utilisateur, la référence d'API et un nouvel AppConfig atelier.	August 3, 2020
prend Configuration rapide désormais en charge AWS Organizations (p. 1409)	prend Configuration rapide désormais en charge la AWS Organizations configuration rapide des rôles de sécurité requis et des Systems Manager fonctionnalités couramment utilisées dans plusieurs comptes et régions. Pour plus d'informations, consultez Configuration rapide d'AWS Systems Manager.	July 23, 2020
Nouvelle source de données dans Explorer : conformité des associations (p. 1409)	affiche Explorer désormais les données de conformité des associations à partir d' State Manager . Pour plus d'informations, consultez Modification des sources de données Systems Manager Explorer.	July 23, 2020
Nouveau document Systems Manager de commande pour activer et désactiverKernel Live Patching (p. 1409)	Le document `AWS-ConfigureKernelLivePatching` est désormais disponible pour une utilisation avec Fonctionnalité Exécuter la commande lorsque vous souhaitez activer ou désactiver Kernel Live Patching sur Amazon Linux 2 des instances . Ce document remplace la nécessité de créer vos propres documents de commande personnalisés pour ces tâches. Pour de plus amples informations, veuillez consulter Use Kernel Live Patching on Amazon Linux 2 instances	July 22, 2020

Quotas Automation mis à jour (p. 1409)	Les quotas de service pour Automation ont été mis à jour, y compris une file d'attente distincte pour les automatisations de contrôle de débit. Pour plus d'informations, consultez AWS Systems Manager Automatisation.	July 20, 2020
Spécifiez le nombre de jours de décalage programmé pour une fenêtre de maintenance à l'aide de la console (p. 1409)	A l'aide de la Systems Manager console, vous pouvez désormais spécifier un nombre de jours à attendre après la date et l'heure spécifiées par une expression CRON avant d'exécuter une fenêtre de maintenance. (Auparavant, cette option était disponible uniquement lorsque vous utilisiez un kit SDK AWS ou un outil de ligne de commande.) Par exemple, si votre expression CRON planifie l'exécution d'une fenêtre de maintenance le troisième mardi de chaque mois à 23h30 – `cron(0 30 23 ? * TUE#3 *)` – et que vous spécifiez un décalage de planification de `2` , la fenêtre ne fonctionnera pas avant deux jours plus tard à 23h30. Pour de plus amples informations, veuillez consulter Cron et expressions de taux pour Systems Manager et spécifier le nombre de jours de décalage programmé pour une fenêtre de maintenance.	July 17, 2020
Mise à jour à PowerShell l'aide d'Fonctionnalité Exécuter la commande (p. 1409)	Pour vous aider à mettre PowerShell à jour vers la version 5.1 sur vos instances Windows Server 2012 et 2012 R2, nous avons ajouté une procédure à .Guide de l'utilisateur AWS Systems Manager Pour plus d'informations, consultez Mise à jour PowerShell à l'aide d' Fonctionnalité Exécuter la commande .	June 30, 2020

prend Gestionnaire de correctifs désormais en charge CentOS 8.0 et 8.1 (p. 1409)	Vous pouvez désormais utiliser Gestionnaire de correctifs pour corriger les instances CentOS 8.0 et 8.1. Pour plus d'informations, consultez les rubriques suivantes : • Sélection des correctifs de sécurité (p. 1049) • Installation des correctifs (p. 1056) • Fonctionnement des règles de référence de correctif sur CentOS (p. 1065) • Installation manuelle de Agent SSM sur les instances CentOS (p. 80) • Installer Agent SSM pour un environnement hybride (Linux) (p. 58)	June 27, 2020

AppConfig s'intègre à AWS CodePipeline. (p. 1409)	AppConfigest une action de déploiement intégrée pour AWS CodePipeline (CodePipeline). CodePipeline est un service de livraison continue entièrement géré qui vous aide à automatiser vos pipelines de publication pour des mises à jour d'application et d'infrastructure rapides et fiables. CodePipeline automatise les phases de génération, de test et de déploiement de votre processus de publication chaque fois qu'un changement de code se produit, en fonction du modèle de publication que vous définissez. L'intégration d' AppConfig à CodePipeline offre les avantages suivants. Pour plus d'informations, consultez AppConfig Intégration à CodePipeline . • Les clients qui utilisent CodePipeline pour gérer l'orchestration disposent désormais d'un moyen léger de déployer des modifications de configuration dans leurs applications sans avoir à déployer la totalité de leur base de code. • Les clients qui souhaitent utiliser AppConfig pour gérer des déploiements de configuration mais sont limités, car AppConfig ne prend pas en charge leur code ou magasin de configuration actuel, ils disposent désormais d'options supplémentaires. CodePipeline prend en charge AWS CodeCommit , et GitHub (pour n'en citer que quelques-unes).BitBucket	June 25, 2020

Réorganisation du chapitre Automation (p. 1409)	Pour vous aider à trouver ce dont vous avez besoin, nous avons réorganisé les rubriques du chapitre Automatisation de .Guide de l'utilisateur AWS Systems Manager Par exemple, les actions Automation et les références aux documents Automation sont désormais des sections de niveau supérieur dans le chapitre. Pour plus d'informations, consultez AWS Systems Manager Automatisation.	June 23, 2020
Nouveau chapitre : Intégrations de produits et services (p. 1409)	Pour vous aider à comprendre comment Systems Manager s'intègre aux services AWS et à d'autres produits et services, un nouveau chapitre a été ajouté à Guide de l'utilisateur AWS Systems Manager . Pour plus d'informations, consultez Intégrations de produits et de services avec Systems Manager .	June 23, 2020
Spécifier le nombre de jours de décalage programmé pour une fenêtre de maintenance (p. 1409)	À l'aide d'un outil de ligne de commande ou d'un kit AWS SDK, vous pouvez désormais spécifier un nombre de jours à attendre après la date et l'heure spécifiées par une expression CRON avant d'exécuter une fenêtre de maintenance. Par exemple, si votre expression CRON planifie l'exécution d'une fenêtre de maintenance le troisième mardi de chaque mois à 23h30 – `cron(0 30 23 ? * TUE#3 *)` – et que vous spécifiez un décalage horaire de `2`, la fenêtre ne s'exécutera que deux jours plus tard à 23h30. Pour de plus amples informations, veuillez consulter Cron et expressions de taux pour Systems Manager et spécifier le nombre de jours de décalage programmé pour une fenêtre de maintenance.	June 19, 2020

pour Kernel Live Patching sur Gestionnaire de correctifs les instancesAmazon Linux 2 (p. 1409)	Kernel Live Patching pour Amazon Linux 2 vous permet d'appliquer des correctifs de vulnérabilité de sécurité et de bogues critiques à un noyau Linux en cours d'exécution, sans redémarrer ni interrompre les applications en cours d'exécution. Vous pouvez désormais activer la fonctionnalité et appliquer des correctifs live du noyau à l'aide d' Gestionnaire de correctifs . Pour plus d'informations, consultez Use Kernel Live Patching on Amazon Linux 2 instances.	June 16, 2020
Gestionnaire de correctifsaugmente Oracle Linux la prise en charge des versions (p. 1409)	Auparavant, ne Gestionnaire de correctifs prenait en charge que la version 7.6 de Oracle Linux . Comme indiqué dans les Gestionnaire de correctifs prérequis, la prise en charge couvre désormais les versions 7.5-7.8.	June 16, 2020
Exemple de scénario pour l'utilisation du paramètre 'InstalloverRidelist' dans AWS-RunPatchBaseline (p. 1409)	La nouvelle rubrique Exemple de scénario d'utilisation du InstallOverrideList paramètre décrit une stratégie d'utilisation du `InstallOverrideList` paramètre dans le `AWS-RunPatchBaseline` document pour appliquer différents types de correctifs à un groupe cible, selon différents calendriers de fenêtre de maintenance, tout en utilisant une seule référence de correctifs.	June 11, 2020
Stratégies de déploiement prédéfinies pour AppConfig (p. 1409)	AppConfig propose désormais des stratégies de déploiement prédéfinies. Pour de plus amples informations, veuillez consulter Création d'un package de déploiement.	June 10, 2020

prend Gestionnaire de correctifs désormais en charge Red Hat Enterprise Linux (RHEL) 7.8-8.2 (p. 1409)	Vous pouvez désormais utiliser Gestionnaire de correctifs pour corriger les instances RHEL 7.8 - 8.2. Pour plus d'informations, consultez les rubriques suivantes : • Sélection des correctifs de sécurité (p. 1049) • Installation des correctifs (p. 1056) • Fonctionnement des règles de référence de correctif sur RHEL (p. 1071) • Installation manuelle de Agent SSM sur les instances Red Hat Enterprise Linux (p. 87) • Installer Agent SSM pour un environnement hybride (Linux) (p. 58)	June 9, 2020
Explorer prend en charge l'administration déléguée (p. 1409)	Si vous agrégez des données Explorerprovenant de plusieurs régions et comptes AWS à l'aide de la synchronisation des données de ressources avec AWS Organizations, nous vous suggérons de configurer un administrateur délégué pour Explorer. Un administrateur délégué améliore la sécurité Explorer en limitant le nombre d'administrateurs Explorer qui peuvent créer ou supprimer des synchronisations de données de ressources de plusieurs comptes et régions à une seule personne. Vous n'avez plus besoin d'être connecté à AWS Organizations compte principal pour administrer la synchronisation des données de ressource dans Explorer . Pour de plus amples informations, veuillez consulter Configuration d'un administrateur délégué.	June 3, 2020

Appliquer l'association State Manager uniquement à l'intervalle Cron spécifié suivant (p. 1409)	Si vous ne souhaitez pas qu'une association State Manager s'exécute immédiatement après sa création, vous pouvez choisir l'option Appliquer l'association uniquement à l'intervalle Cron spécifié suivant dans la console Systems Manager. Pour de plus amples informations, veuillez consulter Créer une association.	June 3, 2020
Nouvelle source de données dans Explorer : AWS Compute Optimizer (p. 1409)	Explorer affiche désormais les données de AWS Compute Optimizer. Cela inclut le nombre d'instances EC2 sous provisionnées et surdimensionnées, les résultats d'optimisation, les détails de tarification à la demande et les recommandations pour le type d'instance et le prix. Pour de plus amples informations, consultez les détails de configuration d'AWS Compute Optimizer dans Configuration des services associés.	May 26, 2020
Nouveau chapitre : Balisage des ressources Systems Manager (p. 1409)	Le nouveau chapitre Balisage des ressources Systems Manager fournit une vue d'ensemble de la façon dont vous pouvez utiliser les balises avec les six types de ressources pouvant être balisées dans Systems Manager. Le chapitre fournit également des instructions complètes pour ajouter et supprimer des balises de ces types de ressources : • Documents • Fenêtres de maintenance • Instances gérées • OpsItems • Parameters • Références de correctifs	May 25, 2020

Installer les Service Packs Windows et les mises à niveau des versions mineures de Linux à l'aide d'Gestionnaire de correctifs (p. 1409)	La nouvelle rubrique – Procédure : Créez une référence de correctifs pour l'installation des Service Packs Windows (console) explique comment créer une référence de correctifs dédiée exclusivement à l'installation des Service Packs Windows. La rubrique Créer une référence de correctifs personnalisée (Linux) a été mise à jour avec des informations concernant l'inclusion des mises à niveau de version mineures pour les systèmes d'exploitation Linux dans les références de correctifs.	May 21, 2020
Réorganisation du chapitre sur Parameter Store (p. 1409)	Toutes les rubriques traitant de la configuration ou de la définition des options pour les opérations Parameter Store ont été regroupées dans la section Mise en route avec Parameter Store. Elle inclut les rubriques Gestion des niveaux de paramètres et Accroissement du débit Parameter Store, qui ont été déplacées depuis d'autres parties du chapitre.	May 18, 2020
À propos des autorisations de chiffrement des SecureString paramètres (p. 1409)	La nouvelle rubrique Autorisations IAM pour l'utilisation des clés par défaut AWS et des clés gérées par le client explique la différence entre le chiffrement de vos paramètres `SecureString` à l'aide d'une clé principale client (CMK) et l'utilisation de la clé AWS KMS par défaut fournie par AWS.	May 13, 2020
Nouvelle rubrique pour créer des chaînes de date et d'heure pour interagir avec les actions d'API Systems Manager. (p. 1409)	La nouvelle rubrique Création de chaînes de date et d'heure formatées pour Systems Manager décrit comment créer des chaînes de date et d'heure formatées pour interagir avec les actions d'API Systems Manager.	May 13, 2020

prend Gestionnaire de correctifs désormais en charge les systèmes d'exploitation Debian Server et Oracle Linux 7.6 (p. 1409)	Vous pouvez désormais utiliser Gestionnaire de correctifs pour corriger Debian Server les Oracle Linux instances et . Gestionnaire de correctifs prend en charge les correctifs des versions Debian Server 8.x, 9.x et Oracle Linux 7.6. Pour de plus amples informations, consultez les rubriques suivantes : • Sélection des correctifs de sécurité (p. 1049) • Installation des correctifs (p. 1056) • Fonctionnement des règles de référence de correctif sur Debian Server (p. 1067) • Fonctionnement des règles de référence de correctif sur Oracle Linux (p. 1069)	May 7, 2020
Créer des associations State Manager qui ciblent Groupes de ressources AWS (p. 1409)	Outre les balises de ciblage, les instances individuelles et toutes les instances de votre compte AWS, vous pouvez désormais créer des associations State Manager qui ciblent des instances dans Groupes de ressources AWS. Pour de plus amples informations, consultez A propos des cibles et des contrôles du débit dans les associations State Manager	May 7, 2020

Nouveau type de données aws:ec2:image dans Parameter Store pour valider l'AMIIDs (p. 1409)	Lorsque vous créez un paramètre `String`, vous pouvez désormais spécifier un type de données comme `aws:ec2:image`, afin de vous assurer que la valeur du paramètre que vous saisissez est un format d'ID d'AMI (Amazon Machine Image) valide. La prise en charge des formats d'ID d'AMI vous permet d'éviter de mettre à jour tous vos scripts et modèles avec un nouvel ID lors de chaque changement de l'AMI que vous souhaitez utiliser dans vos processus. Vous pouvez créer un paramètre avec le type de données `aws:ec2:image`, et saisir pour sa valeur l'ID d'une AMI. Il s'agit de l'AMI à partir de laquelle vous souhaitez actuellement créer de nouvelles instances. Vous référencez ensuite ce paramètre dans vos modèles, commandes. Lorsque vous êtes prêt à utiliser une autre AMI, mettez à jour la valeur du paramètre. Parameter Store valide le nouvel ID d'AMI et vous n'avez pas besoin de mettre à jour vos scripts et modèles. Pour plus d'informations, consultez Prise en charge des paramètres natifs pour Amazon Machine ImageIDs.	May 5, 2020
Gestion des codes de sortie dans les commandes Exécuter la commande (p. 1409)	La fonctionnalité Exécuter la commande vous permet de définir la manière dont les codes de sortie sont gérés dans vos scripts. Par défaut, le code de sortie de la dernière commande exécutée dans un script est signalé comme le code de sortie pour l'ensemble du script. Cependant, vous pouvez inclure une instruction conditionnelle shell pour quitter le script si une commande précédant la dernière échoue à l'aide de l'approche suivante. Pour obtenir des exemples, consultez la nouvelle rubrique Gestion des codes de sortie dans les commandes Fonctionnalité Exécuter la commande.	May 5, 2020

Nouvelle source de données dans Explorer : AWS Trusted Advisor (p. 1409)	Explorer affiche désormais les données de AWS Trusted Advisor. Cela inclut l'état des vérifications des bonnes pratiques et les recommandations dans les domaines suivants : optimisation des coûts, sécurité, tolérance aux pannes, performances et limites de service. Pour de plus amples informations, consultez les détails de configuration d'Trusted Advisor dans Configuration des services associés.	May 4, 2020
Nouveaux paramètres publics publiés pour les zones de disponibilité et les zones locales (p. 1409)	Des paramètres publics ont été publiés pour rendre les informations sur les zones de disponibilitéet les zones locales AWS disponibles par programmation. Ces paramètres s'ajoutent aux paramètres publics de l'infrastructure mondiale existants pour les services et les régions AWS. Pour de plus amples informations, consultez Appeler les paramètres publics du service, de la région, du point de terminaison, de la zone de disponibilité et de la zone locale AWS.	May 4, 2020
Nouvelle rubrique pour les quotas et les limitations d'un magasin de configurations AppConfig (p. 1409)	Nous avons ajouté une nouvelle rubrique pour aider les utilisateurs à comprendre rapidement les quotas de service et les limitations de chaque magasin de configurations AppConfig. Pour de plus amples informations, veuillez consulter À propos des quotas et des limitations d'un magasin de configurations.	April 17, 2020

Afficher les Amazon CloudWatch alarmes pour Groupes de ressources AWS (p. 1409)	Outre les quatre onglets de données d'opérations ajoutés à la page Groupes de ressources AWS dans Systems Manager plus tôt ce mois-ci, la page Groupes de ressources AWS inclut désormais un onglet Monitoring (Surveillance) . Cet onglet affiche les alarmes CloudWatch et les tableaux de bord des ressources du groupe sélectionné. Ces onglets de données d'opérations ne sont pas disponibles lors de l'affichage d'un groupe dans la console du service Groupes de ressources AWS. Les informations de ces onglets peuvent vous aider à comprendre quelles ressources au sein d'un groupe sont conformes et fonctionnent correctement, et quelles ressources demandent une action. Si vous devez agir sur une ressource, vous pouvez utiliser les runbooks d'automatisation Systems Manager pour effectuer des tâches courantes de maintenance et de dépannage. Pour de plus amples informations, veuillez consulter Affichage des données d'opérations pour Groupes de ressources AWS.	March 25, 2020

Créer des associations State Manager qui exécutent des recettes Chef (p. 1409)	Vous pouvez créer des associations State Manager qui exécutent des livres de recettes et des recettes Chef à l'aide du document AWS-ApplyChefRecipes. Ce document offre les avantages suivants pour l'exécution de recettes Chef : • Il prend en charge plusieurs versions de Chef (Chef 11 à Chef 14). • Il installe automatiquement le logiciel client Chef sur les instances cibles. • Exécute éventuellement des contrôles de conformité Systems Manager sur les instances cibles et stocke les résultats des contrôles de conformité dans un compartiment S3. • Il exécute plusieurs livres de cuisine et recettes en une seule fois du document. • Il peut exécuter des recettes en mode `why-run`, pour afficher lesquelles changeront sur les instances cibles sans y apporter de modifications. • Il peut appliquer des attributs JSON personnalisés aux exécutions `chef-client`. Pour de plus amples informations, veuillez consulter Création d'associations qui exécutent des recettes Chef.	March 19, 2020
Mise à jour de la rubrique « Groupes de ressources » (p. 1409)	La rubrique Groupes de ressources AWS a été mise à jour pour fournir des informations plus complètes sur l'utilisation des groupes de ressources dans Systems Manager, et sur l'utilisation des groupes de ressources dans vos opérations Systems Manager.	March 17, 2020

Synchroniser les données d'inventaire de plusieurs comptes AWS vers un compartiment Amazon S3 central (p. 1409)	Vous pouvez synchroniser les données d'inventaire Systems Manager de plusieurs comptes AWS vers un compartiment central S3. Les comptes doivent être définis dans AWS Organizations. Pour de plus amples informations, veuillez consulter Création d'une synchronisation des données de ressources d'inventaire pour plusieurs comptes définis dans des organisations AWS.	March 16, 2020
Afficher les données d'opérations pour Groupes de ressources AWS (p. 1409)	La Groupes de ressources AWS page dans affiche Systems Manager désormais les données d'opérations d'un groupe sélectionné sur quatre onglets : Détails, Config, CloudTrail , OpsItems . Ces onglets ne sont pas disponibles lors de l'affichage d'un groupe dans la console Groupes de ressources AWS de service. Les informations de ces onglets peuvent vous aider à comprendre quelles ressources au sein d'un groupe sont conformes et fonctionnent correctement, et quelles ressources demandent une action. Si vous devez agir sur une ressource, vous pouvez utiliser les runbooks d'automatisation Systems Manager pour effectuer des tâches courantes de maintenance et de dépannage. Pour de plus amples informations, veuillez consulter Affichage des données d'opérations pour Groupes de ressources AWS.	March 16, 2020
Stocker des configurations AppConfig dans Amazon S3 (p. 1409)	Auparavant, AppConfig prenait uniquement en charge des configurations d'applications stockées dans des documents Systems Manager (SSM) ou des paramètres Parameter Store. Outre ces options, AppConfig prend désormais en charge le stockage des configurations dans Amazon S3. Pour de plus amples informations, veuillez consulter À propos des configurations stockées dans Amazon S3.	March 13, 2020

Agent SSMinstallé par défaut sur Amazon ECS-optimizedAMIs (p. 1409)	Agent SSMest désormais installé par défaut sur les optimisées pour Amazon ECSAMIs. Pour plus d'informations, consultez Utilisation d' Agent SSM .	February 25, 2020
Créer des configurations AppConfig dans la console (p. 1409)	AppConfig vous permet désormais de créer une configuration d'application dans la console au moment de la création d'un profil de configuration. Pour de plus amples informations, veuillez consulter Créer une configuration et un profil de configuration.	February 13, 2020
Approuver automatiquement uniquement les correctifs publiés jusqu'à une date spécifiée (p. 1409)	Outre l'option permettant d'approuver automatiquement les correctifs pour l'installation un nombre spécifié de jours après leur publication, prend Gestionnaire de correctifs désormais en charge la possibilité d'approuver automatiquement uniquement les correctifs publiés à une date que vous spécifiez ou avant. Par exemple, si vous spécifiez le 7 juillet 2020 comme date limite dans votre référence de correctifs, aucun correctif publié à partir du 8 juillet 2020 n'est installé automatiquement. Pour de plus amples informations, veuillez consulter À propos des références personnalisées et Créer une référence de correctifs personnalisée.	February 12, 2020

Utiliser le pseudo-paramètre {{RESOURCE_ID}} dans les tâches de la fenêtre de maintenance (p. 1409)	Lorsque vous enregistrez une tâche de fenêtre de maintenance, vous spécifiez les paramètres uniques au type de tâche. Vous pouvez référencer certaines valeurs à l'aide de la syntaxe des pseudo-paramètres, comme `{{TARGET_ID}}` , et `{{TARGET_TYPE}}` .`{{WINDOW_TARGET_ID}}` Quand la tâche de fenêtre de maintenance s'exécute, elle transmet les valeurs correctes au lieu des espaces réservés des pseudo-paramètres. Pour prendre en charge les ressources qui font partie d'un groupe de ressources en tant que cible, vous pouvez utiliser le pseudo-paramètre `{{RESOURCE_ID}}` afin de transmettre des valeurs pour des ressources telles que des tables DynamoDB, des compartiments S3 et d'autres types pris en charge. Pour plus d'informations, consultez les rubriques suivantes dans le didacticiel : Créez et configurez une fenêtre de maintenance (AWS CLI) : • À propos des pseudo-paramètres • Exemples : Enregistrer les tâches avec une fenêtre de maintenance	February 6, 2020
Réexécuter rapidement les commandes (p. 1409)	Systems Manager comprend deux options pour vous aider à réexécuter une commande à partir de la page Exécuter la commande de la console AWS Systems Manager. Réexécuter : Ce bouton vous permet d'exécuter la même commande sans y apporter de modifications. Copier vers le nouveau : Ce bouton copie les paramètres d'une commande dans une nouvelle commande et vous donne la possibilité de modifier ces paramètres avant de l'exécuter. Pour de plus amples informations, veuillez consulter Réexécution des commandes.	February 5, 2020

Revenir du niveau des instances avancées au niveau des instances standard (p. 1409)	Si vous avez précédemment configuré toutes les instances locales exécutées dans votre environnement hybride pour utiliser le niveau des instances avancées, vous pouvez désormais configurer rapidement ces instances pour qu'elles utilisent le niveau des instances standard. Le retour au niveau des instances standard s'applique à toutes les instances hybrides d'un compte AWS et d'une seule région AWS. Le retour au niveau des instances standard a une incidence sur la disponibilité de certaines fonctionnalités Systems Manager. Pour de plus amples informations, veuillez consulter Retour du niveau des instances avancées au niveau des instances standard	January 16, 2020
Nouvelle option pour ignorer les redémarrages d'instance après l'installation du correctif (p. 1409)	Auparavant, les instances gérées étaient toujours redémarrées après lGestionnaire de correctifs'installation des correctifs sur celles-ci. Un nouveau paramètre `RebootOption` dans le document SSM `AWS-RunPatchBaseline` vous permet de spécifier si vous souhaitez ou non que vos instances redémarrent automatiquement après l'installation de nouveaux correctifs. Pour plus d'informations, consultez Nom du paramètre : RebootOptiondans la rubrique À propos du document SSM AWS-RunPatchBaseline.	January 15, 2020
Nouvelle rubrique : « Schéma et syntaxe du document d'automatisation » (p. 1409)	Des exemples et des informations sur les éléments de Schema 0.3, qui est utilisé par les documents SSM Automation, sont désormais disponibles dans la rubrique Schéma et syntaxe du document Automation.	January 10, 2020

Nouvelle rubrique : 'Exécution PowerShell de scripts sur des instances Linux' (p. 1409)	Nouvelle rubrique qui décrit comment utiliser la fonctionnalité Exécuter la commande pour exécuter PowerShell des scripts sur des instances Linux. Pour plus d'informations, consultez Exécution PowerShell de scripts sur des instances Linux.	January 10, 2020
Mises à jour de « Configurer Agent SSM afin d'utiliser un proxy » (p. 1409)	Les valeurs à spécifier lors de la configuration de Agent SSM pour utiliser un proxy ont été mises à jour pour refléter les options des serveurs proxy HTTP et HTTPS. Pour de plus amples informations, veuillez consulter Configurer Agent SSM afin d'utiliser un proxy.	January 9, 2020
Nouveau chapitre « Sécurité » décrivant les pratiques de sécurisation des ressources Systems Manager (p. 1409)	Un nouveau chapitre Sécurité dans le Guide de l'utilisateur AWS Systems Manager vous aide à comprendre comment appliquer le modèle de responsabilité partagée lorsque vous utilisez Systems Manager. Les rubriques de ce chapitre vous montrent comment configurer Systems Manager pour atteindre vos objectifs de sécurité et de conformité. Vous pouvez également apprendre à utiliser d'autres services AWS capables de vous aider à surveiller et à sécuriser vos ressources Systems Manager. Note Dans le cadre de cette mise à jour, le chapitre « Authentification et contrôle d'accès » du guide de l'utilisateur a été remplacé par une nouvelle section plus simple, Identity and access management (IAM) pour AWS Systems Manager.	December 24, 2019

Nouveaux exemples de documents Automatisation personnalisés (p. 1409)	Un ensemble d'exemples de documents Automation personnalisés a été ajouté au guide de l'utilisateur. Ces exemples montrent comment utiliser diverses actions Automation pour simplifier les tâches de déploiement, de dépannage et de maintenance, et sont destinés à vous aider à écrire vos propres documents Automation personnalisés. Pour de plus amples informations, veuillez consulter Exemples de documents Automation personnalisés. Vous pouvez également afficher le contenu des documents Automation gérés par Amazon dans la console Systems Manager. Pour de plus amples informations, veuillez consulter Référence sur les détails des documents Systems Manager Automation.	December 23, 2019
Prise en charge de l'Oracle Linux (p. 1409)	prend Systems Manager désormais en charge Oracle Linux 7.5 et 7.7. Pour plus d'informations sur l'installation manuelle Agent SSM de sur les instances EC2, consultez Oracle Linux Oracle Linux . Pour plus d'informations sur l'installation Agent SSM de sur Oracle Linux des serveurs dans un environnement hybride, consultez l'étape 6 : Installez l'agent SSM pour un environnement hybride (Linux).	December 19, 2019

Possibilité d'ajouter un script à votre flux de travail à partir d'un autre document Automation (p. 1409)	Lorsque vous créez un document Automation à l'aide d'un outil de ligne de commande, vous pouvez désormais ajouter un script qui est déjà utilisé dans un autre document à une étape de votre nouveau document. Des scripts sont utilisés avec le type `aws:executeScript` d'action. Vous pouvez ajouter des scripts à partir de documents Automation qui vous appartiennent ou qui sont partagés avec vous à partir d'un autre compte AWS. Pour de plus amples informations, veuillez consulter Création d'un document Automation exécutant des scripts (ligne de commande).	December 19, 2019
Lancer les sessions Session Manager à partir de la console Amazon EC2 (p. 1409)	Vous pouvez désormais démarrer des sessions Session Manager à partir de la console Amazon Elastic Compute Cloud (Amazon EC2). L'utilisation de tâches liées à la session à partir de la console Amazon EC2 nécessite des autorisations IAM différentes aussi bien pour les utilisateurs que pour les administrateurs. Vous pouvez fournir des autorisations pour l'utilisation de la consoleSession Manager et de l'AWS CLI uniquement, pour l'utilisation de la console Amazon EC2 uniquement, ou pour l'utilisation des trois outils. Pour plus d'informations, consultez les rubriques suivantes. • Démarrage rapide - Stratégies IAM par défaut pour Session Manager • Démarrage d'une session (console Amazon EC2)	December 18, 2019

Prise en charge des métriques et des alarmes Fonctionnalité Exécuter la commande par CloudWatch (p. 1409)	AWS Systems Manager transmet désormais des métriques sur le statut des commandes Fonctionnalité Exécuter la commande à CloudWatch, ce qui vous permet de définir des alarmes en fonction de ces métriques. Les valeurs de statut terminal des commandes pour lesquelles vous pouvez suivre les métriques incluent `Success` , et `Failed` .`Delivery Timed Out` Pour plus d'informations, consultez Surveillance Fonctionnalité Exécuter la commande des métriques à l'aide d' Amazon CloudWatch .	December 17, 2019
Nouvelle Systems Manager fonctionnalité : Change Calendar (p. 1409)	Utilisez Systems Manager Change Calendar pour spécifier des périodes (événements) pendant lesquelles vous souhaitez limiter ou empêcher les modifications de code (par exemple, à partir de documents Systems Manager Automation ou de fonctions AWS Lambda) dans les ressources. Un nouveau type de Systems Manager document Change Calendar stocke les données iCalendar 2.0 au format texte brut. Pour de plus amples informations, veuillez consulter Modifier le calendrier AWS Systems Manager.	December 11, 2019

Nouvelle Systems Manager fonctionnalité : AWS AppConfig (p. 1409)	AppConfig permet de créer, gérer et déployer rapidement des configurations d'applications. AppConfig prend en charge les déploiements contrôlés vers des applications de toute taille. Vous pouvez utiliser AppConfig avec des applications hébergées sur des instances EC2, AWS Lambda , des conteneurs, des applications mobiles ou des IoT appareils. Pour éviter les erreurs lors du déploiement de configurations d'application, AppConfig propose des validateurs. Un validateur permet une vérification syntaxique ou sémantique pour s'assurer que la configuration que vous souhaitez déployer fonctionne comme prévu. Lors d'un déploiement de configuration, AppConfig surveille l'application pour s'assurer que le déploiement a réussi. Si le système rencontre une erreur ou si le déploiement déclenche une alarme, AppConfig annule la modification afin de minimiser l'impact sur les utilisateurs de votre application. Pour de plus amples informations, veuillez consulter AWS AppConfig.	November 25, 2019
Modifications de la navigation à gauche dans la console Systems Manager (p. 1409)	Vous pouvez désormais trouver les groupes AWS Resource Groups et Parameter Store sous Application Management. Pour voir la documentation relative à ces fonctionnalités, veuillez consulter AWS Systems Manager Application Management.	November 25, 2019

Nouvelle fonctionnalité Systems Manager : Systems Manager Explorer (p. 1409)	AWS Systems Manager Explorer est un tableau de bord des opérations personnalisable qui présente des informations sur vos ressources AWS. Explorer affiche une vue agrégée des données d'opérations (OpsData) pour vos comptes et pour toutes les régions AWS. Dans Explorer , OpsData inclut des métadonnées sur vos instances EC2, des détails de conformité des correctifs et des éléments de travail opérationnels (OpsItems). Explorer fournit un contexte sur la répartition entre vos unités commerciales ou applications, leur évolution au fil du temps et leur variation par catégorie.OpsItems Vous pouvez regrouper et filtrer les informations dans Explorer pour vous concentrer sur les éléments qui vous intéressent et qui nécessitent une action. Lorsque vous identifiez des problèmes prioritaires, vous pouvez utiliser Systems Manager OpsCenter pour exécuter des runbooks Automation et résoudre rapidement ces problèmes. Pour de plus amples informations, veuillez consulter AWS Systems ManagerExplorer. Note La configuration pour Systems Manager OpsCenter est intégrée à la configuration pour Explorer. Si vous avez déjà configuré OpsCenter, vous devez tout de même terminer l'installation intégrée pour vérifier les paramètres et les options. Si vous n'avez pas configuré OpsCenter, vous pouvez utiliser l'installation intégrée pour commencer avec les deux fonctions. Pour de plus amples informations, veuillez	November 18, 2019

	consulter Démarrer avec Explorer et OpsCenter.	
Amélioration des capacités de recherche de paramètres (p. 1409)	Les outils de recherche de paramètres facilitent désormais la recherche de paramètres lorsque vous en avez un grand nombre dans votre compte ou lorsque vous avez oublié le nom exact d'un paramètre. L'outil de recherche vous permet désormais de filtrer par `contains` . Auparavant, les outils de recherche prenaient uniquement en charge la recherche de noms de paramètres par `equals` et `begins-with` . Pour plus d'informations, consultez Recherche de Systems Manager paramètres.	November 15, 2019

Nouveau Document Builder for Automation basé sur la console | Prise en charge de l'exécution de scripts dans les étapes Automation (p. 1409)	Vous pouvez désormais utiliser Systems Manager Automation pour créer et partager des playbooks opérationnels standardisés afin de garantir la cohérence entre les utilisateurs, les comptes et les régions AWS. Grâce à la possibilité d'exécuter des scripts et d'ajouter de la documentation en ligne à vos documents Automation à l'aide de Markdown, vous pouvez réduire les erreurs et éliminer les étapes manuelles telles que la navigation dans les procédures écrites dans les wikis et l'exécution de commandes de terminal. Pour plus d'informations, consultez les rubriques suivantes. • Procédure : Utilisation de Document Builder pour créer un document Automation poussiéreux • Documents Automation gérés par Amazon qui exécutent des scripts • aws:executeScript (Référence sur les actions d'automatisation) • Création de documents Automation qui exécutent des scripts • Création de documents Automation à l'aide de Document Builder • Nouvelles fonctionnalités d'automatisation dans Systems Manager sur le blog AWS News	November 14, 2019

Effectuer une mise à jour de package sur place à l'aide de Distributor (p. 1409)	Auparavant, lorsque vous vouliez installer une mise à jour sur un package en utilisant Distributor, votre seul choix consistait à désinstaller le package entier et à réinstaller la nouvelle version. Maintenant, vous pouvez choisir d'effectuer une mise à jour sur place au lieu de cela. Lors d'une mise à jour sur place, Distributor installe uniquement les fichiers nouveaux ou modifiés depuis la dernière installation, conformément au script de mise à jour que vous incluez dans votre package. Avec cette option, votre application de package peut rester disponible et ne pas être mise hors connexion pendant la mise à jour. Pour plus d'informations, consultez les rubriques suivantes. • Créer un package • Installer ou mettre à jour des packages	November 11, 2019
Nouvelle fonction de mise à jour automatique Agent SSM (p. 1409)	En un seul clic, vous pouvez configurer toutes les instances de votre compte AWS pour rechercher et télécharger automatiquement de nouvelles versions de Agent SSM. Pour ce faire, choisissez Agent auto update (Mise à jour automatique de l'agent) dans la page Managed instances (Instances gérées) de la console AWS Systems Manager. Pour de plus amples informations, veuillez consulter Automatiser les mises à jour vers Agent SSM.	November 5, 2019

Restreindre l'accès Session Manager à l'aide de balises fournies par AWS (p. 1409)	Une deuxième méthode de contrôle de l'accès des utilisateurs aux actions de session est désormais disponible. Cette nouvelle méthode vous permet de créer des stratégies d'accès IAM à l'aide de balises de session fournies par AWS au lieu d'utiliser la variable `{aws:username}`. L'utilisation de ces balises de session fournies par AWS permet aux organisations qui utilisent la fédération de contrôler l'accès des utilisateurs aux sessions.IDs Pour de plus amples informations, veuillez consulter Autoriser un utilisateur à interrompre uniquement les sessions qu'il a démarrées.	October 2, 2019
Nouveau document SSM Ansible : Manuels AWS-ApplyAnsible (p. 1409)	Vous pouvez créer des associations State Manager qui exécutent des manuels stratégiques Ansible à l'aide du document AWS-ApplyAnsiblePlaybooks. Ce document offre les avantages suivants pour l'exécution de manuels stratégiques : • Prise en charge de l'exécution de manuels stratégiques complexes • Prise en charge du téléchargement de manuels stratégiques depuis GitHub et Amazon Simple Storage Service (Amazon S3) • Prise en charge de la structure de manuel stratégique compressé • Journalisation améliorée • Possibilité de spécifier le manuel stratégique à exécuter lorsque les manuels stratégiques sont regroupés Pour de plus amples informations, veuillez consulter Création d'associations exécutant des manuels stratégiques Ansible.	September 24, 2019

Prise en charge du réacheminement de port pour Session Manager (p. 1409)	Session Manager prend désormais en charge les sessions de réacheminement de port. Le réacheminement de port vous permet de créer en toute sécurité des tunnels entre vos instances déployées dans des sous-réseaux privés, sans avoir besoin de démarrer le service SSH sur le serveur, d'ouvrir le port SSH dans le groupe de sécurité ou d'utiliser un hôte bastion. Comme pour les tunnels SSH, le réacheminement de port vous permet de réacheminer le trafic entre votre ordinateur portable pour ouvrir des ports sur votre instance. Une fois le réacheminement de port configuré, vous pouvez vous connecter au port local et accéder à l'application serveur qui s'exécute sur l'instance. Pour de plus amples informations, consultez les rubriques suivantes : • Port Forwarding Using AWS Systems Manager Session Manager on the AWS News Blog • Démarrage d'une session (réacheminement de port)	August 29, 2019

Spécifiez un niveau de paramètre par défaut ou automatisez la sélection de niveau (p. 1409)	Vous pouvez désormais spécifier un niveau de paramètre par défaut qui sera utilisé pour les demandes de création ou de mise à jour d'un paramètre ne spécifiant pas de niveau. Vous pouvez définir le niveau par défaut sur des paramètres standard, des paramètres avancés ou une nouvelle option, Intelligent-Tiering (Hiérarchisation intelligente). Intelligent-Tiering évalue chaque PutParameter demande et crée un paramètre avancé uniquement lorsque cela est nécessaire. (Les paramètres avancés sont obligatoires si la taille de la valeur du paramètre est supérieure à 4 Ko, si une stratégie de paramètre est associée au paramètre ou si les 10 000 paramètres maximum pris en charge pour le niveau standard sont déjà créés.) Pour plus d'informations sur la spécification d'un niveau par défaut et l'utilisation d'Intelligent-Tiering, consultez Spécification d'un niveau de paramètre par défaut.	August 27, 2019
Section Utilisation d'associations mise à jour avec l'interface de ligne de commande et PowerShell les procédures (p. 1409)	La section Utilisation d'associations a été mise à jour en incluant la documentation procédurale relative à la gestion des associations à l'aide de l'AWS CLI ou d'Outils AWS pour PowerShell. Pour de plus amples informations, veuillez consulter Utilisation d'associations dans Systems Manager.	August 26, 2019
Mise à jour de la section Utilisation des exécutions Automation avec l'interface de ligne de commande et PowerShell les procédures (p. 1409)	La section Utilisation des exécutions Automation a été mise à jour en incluant la documentation procédurale relative à l'exécution des flux de travail Automation à l'aide de l'AWS CLI ou d'Outils AWS pour PowerShell. Pour de plus amples informations, veuillez consulter Utilisation des exécutions Automation.	August 20, 2019

OpsCenter est intégré à Application Insights (p. 1409)	OpsCenter est intégré à Amazon CloudWatch Application Insights pour .NET et SQL Server. Cela signifie que vous pouvez créer automatiquement des éléments OpsItems pour les problèmes détectés dans vos applications. Pour de plus amples informations sur la configuration d'Application Insights pour créer des OpsItems, veuillez consulter Configuration de votre application dans le Guide de l'utilisateur Amazon CloudWatch.	August 7, 2019
Nouvelle fonction de console :AWS Systems Manager Configuration rapide (p. 1409)	La configuration rapide est une nouvelle fonctionnalité de la console Systems Manager qui vous permet de configurer rapidement plusieurs composants Systems Manager sur vos instances EC2. Plus précisément, la configuration rapide vous aide à configurer les composants suivants sur les instances que vous choisissez ou ciblez à l'aide de balises : • Un rôle de profil d'instance AWS Identity and Access Management (IAM) pour Systems Manager. • Une mise à jour planifiée bimensuelle d'Agent SSM. • Une collecte planifiée de métadonnées d'inventaire toutes les 30 minutes. • Une analyse quotidienne de vos instances pour identifier les correctifs manquants. • Une installation et une configuration uniques de l'agent Amazon CloudWatch. • Une mise à jour mensuelle planifiée de l'agent CloudWatch. Pour de plus amples informations, veuillez consulter Configuration rapide AWS Systems Manager.	August 7, 2019

Enregistrer un groupe de ressources en tant que cible d'une fenêtre de maintenance (p. 1409)	En plus d'enregistrer des instances gérées comme cible d'une fenêtre de maintenance, vous pouvez désormais enregistrer un groupe de ressources pour cela. Fenêtres de maintenance prend en charge tous les types de ressources AWS qui sont pris en charge par Groupes de ressources AWS, notamment AWS::EC2::Instance, AWS::DynamoDB::Table, AWS::OpsWorks::Instance, AWS::Redshift::Cluster, et plus encore. Avec cette version, vous pouvez également envoyer des commandes à un groupe de ressources, par exemple à l'aide de la console Fonctionnalité Exécuter la commande, de l'AWS CLI ou de la commande send-command. Pour de plus amples informations, consultez les rubriques suivantes : • Affecter des cibles à une fenêtre de maintenance (console) • Exemples : Enregistrement des cibles avec une fenêtre de maintenance • Utilisation des contrôles de cibles et de taux pour envoyer des commandes à un parc	July 23, 2019
Simplification de la création de package et de la gestion des versions de AWS Systems Manager Distributor (p. 1409)	Distributor comporte un nouveau flux de travail de création de package simplifié qui peut générer un manifeste de package, des scripts et des hachages de fichier pour vous. Vous pouvez également utiliser le flux de travail simplifié lorsque vous ajoutez une version à un package existant.	July 22, 2019
Nouveau panneau de catégories de document pour Systems Manager Automation (p. 1409)	Systems Manager inclut un nouveau panneau de catégories de document lorsque vous exécutez une automatisation dans la console. Utilisez ce panneau pour filtrer les documents d'automatisation en fonction de leur finalité.	July 18, 2019

Vérification des autorisations utilisateur pour l'accès au document de configuration Session Manager par défaut (p. 1409)	Lorsqu'un utilisateur de votre compte utilise l' AWS CLI pour démarrer une Session Manager session et ne spécifie pas de document de configuration dans la commande, Systems Manager utilise le document de configuration par défaut`SSM-SessionManagerRunShell`. Vous pouvez maintenant vérifier que l'utilisateur a reçu l'autorisation d'accéder à ce document en ajoutant un élément de condition pour `ssm:SessionDocumentAccessCheck` à la stratégie de l'utilisateur IAM. Pour de plus amples informations, veuillez consulter Application de la vérification d'autorisation de document pour le scénario d'interface de ligne de commande par défaut.	July 9, 2019
Prise en charge du démarrage de sessions Session Manager à l'aide des informations d'identification de l'utilisateur de système d'exploitation (p. 1409)	Par défaut, les sessions Session Manager sont lancées à l'aide des informations d'identification d'un compte `ssm-user` généré par le système qui est créé sur une instance gérée. Sur les machines Linux, vous pouvez désormais plutôt lancer des sessions en utilisant les informations d'identification d'un compte de système d'exploitation. Pour de plus amples informations, veuillez consulter Activer la prise en charge de Run As pour les instances Linux.	July 9, 2019
Prise en charge du démarrage de sessions Session Manager à l'aide de SSH (p. 1409)	Vous pouvez désormais utiliser l'AWS CLI pour démarrer une session SSH sur une instance gérée à l'aide de Session Manager. Pour de plus amples informations sur l'activation de sessions SSH avec Session Manager, veuillez consulter (Facultatif) Activer des sessions SSH Session Manager. Pour de plus amples informations sur le lancement d'une session SSH avec Session Manager, veuillez consulter Démarrage d'une session (SSH).	July 9, 2019

Prise en charge de la modification des mots de passe sur les instances gérées (p. 1409)	Vous pouvez maintenant réinitialiser des mots de passe sur des machines que vous gérez avec Systems Manager (instances gérées). Vous pouvez réinitialiser le mot de passe à l'aide de la console Systems Manager ou de l'AWS CLI. Pour de plus amples informations, veuillez consulter Réinitialisation de mots de passe sur des instances gérées.	July 9, 2019
Révisions apportées à « Qu'est-ce qu'AWS Systems Manager? » (p. 1409)	Le contenu de présentation Qu'est-ce qu' AWS Systems Manager? a été étendu pour présenter le service plus en détails et refléter les fonctionnalités Systems Manager qui ont été lancées récemment. En outre, d'autres contenus de la section ont été déplacés vers des rubriques individuelles pour offrir une meilleure visibilité.	June 10, 2019

Nouvelle fonctionnalité Systems Manager : OpsCenter (p. 1409)	OpsCenter offre un emplacement centralisé où les ingénieurs d'exploitation et les professionnels de l'informatique peuvent consulter, étudier et résoudre des éléments de travail opérationnels (OpsItems) associés aux ressources AWS. OpsCenter est conçu pour réduire la durée moyenne de résolution des problèmes ayant un impact sur les ressources AWS. Cette fonctionnalité Systems Manager regroupe et normalise les OpsItems entre les services tout en fournissant des données d'investigation contextuelles sur chaque OpsItem, sur les OpsItems associés et sur les ressources connexes. OpsCenter fournit également des documents Automation Systems Manager (runbooks) que vous pouvez utiliser pour résoudre rapidement les problèmes. Vous pouvez spécifier des données personnalisées consultables pour chaque OpsItem. Vous pouvez également afficher des rapports récapitulatifs sur les OpsItems générés automatiquement, par statut et par source. Pour plus d'informations, consultez AWS Systems ManagerOpsCenter.	June 6, 2019
Modifications apportées au panneau de navigation de gauche de Systems Manager dans la AWS Management Console (p. 1409)	Le panneau de navigation de gauche de Systems Manager dans la AWS Management Console inclut de nouvelles rubriques, notamment une nouvelle rubrique pour le centre d'opérations, qui fournissent un regroupement logique de fonctionnalités Systems Manager.	June 6, 2019

Révision du didacticiel pour la création et la configuration d'une fenêtre de maintenance à l'aide de l'AWS CLI (p. 1409)	Didacticiel : La création et la configuration d'une fenêtre de maintenance (AWS CLI) ont été réorganisées afin de fournir un chemin simple tout au long des étapes pratiques. Vous pouvez créer une fenêtre de maintenance simple, identifier une cible unique et configurer une tâche simple pour la fenêtre de maintenance à exécuter. Dans le même temps, nous fournissons des informations et des exemples que vous pouvez utiliser pour créer vos propres commandes d'enregistrement des tâches, y compris les informations d'utilisation des pseudo-paramètres tels que `{{TARGET_ID}}` . Pour plus d'informations et d'exemples, consultez les rubriques suivantes : • Exemples : Enregistrement des cibles avec une fenêtre de maintenance • Exemples : Enregistrer les tâches avec une fenêtre de maintenance • À propos des options register-task-with-maintenance-windows • À propos des pseudo-paramètres	May 31, 2019
Notifications sur les mises à jour de Agent SSM (p. 1409)	Pour recevoir des notifications sur les Agent SSM mises à jour, abonnez-vous à la page Agent SSM Notes de mise à jour sur GitHub .	May 24, 2019

Réception de notifications ou déclenchement d'actions en fonction des modifications apportées à Parameter Store (p. 1409)	La rubrique Configuration de notifications ou déclenchement d'actions basées sur les événements Parameter Store vous aide à configurer des règles Amazon EventBridge permettant de suivre les modifications apportées à Parameter Store. Vous pouvez recevoir des notifications ou déclencher d'autres actions lorsque l'une des situations suivantes se produit : • Un paramètre est créé, mis à jour ou supprimé. • La version d'une étiquette de paramètre est créée, mise à jour ou supprimée. • Un paramètre expire, est sur le point d'expirer ou n'a pas été modifié au cours de la période spécifiée.	May 22, 2019

Révisions majeures du contenu des chapitres Configuration et Mise en route (p. 1409)	Nous avons développé et réorganisé le contenu des chapitres Configuration et Mise en route dans le Guide de l'utilisateur d'AWS Systems Manager. Le contenu du chapitre Configuration a été divisé en deux sections. Une section se concentre sur les tâches de configuration de Systems Manager pour configurer et gérer vos instances EC2. Les autres sections traitent des tâches de configuration de Systems Manager pour configurer et gérer vos serveurs sur site et vos machines virtuelles dans un environnement hybride. Les deux sections contiennent désormais toutes les rubriques de configuration sous la forme d'étapes principales numérotées, classées par ordre d'achèvement recommandé. Le nouveau chapitre Mise en route a pour objectif d'aider les utilisateurs finaux à démarrer avec Systems Manager après avoir terminé les tâches de configuration de compte et de service. • Configuration de AWS Systems Manager • Configuration d'AWS Systems Manager pour les environnements hybrides • Mise en route avec AWS Systems Manager	May 15, 2019

Inclusion de correctifs pour les applications Microsoft dans les références de correctifs (Windows) (p. 1409)	prend Gestionnaire de correctifs désormais en charge les mises à jour de correctifs pour les applications Microsoft sur les Windows Server instances . Auparavant, seuls les correctifs pour le système Windows Server d'exploitation étaient pris en charge. Gestionnaire de correctifs fournit deux références de correctifs prédéfinies pour les Windows Server instances. La référence de correctifs `AWS-WindowsPredefinedPatchBaseline-OS` s'applique uniquement aux correctifs du système d'exploitation. `AWS-WindowsPredefinedPatchBaseline-OS-Applications`s'applique au système Windows Server d'exploitation et aux applications Microsoft sous Windows. Pour de plus amples informations sur la création d'une référence de correctifs personnalisée qui inclut des correctifs pour les applications Microsoft, veuillez consulter la première procédure dans Création d'une référence de correctifs personnalisée. De plus, dans le cadre de cette mise à jour, les noms des références de correctifs prédéfinies fournies par AWS sont modifiés. Pour de plus amples d'informations, veuillez consulter Références prédéfinies.	May 7, 2019
Exemples d'enregistrement de cibles de fenêtre de maintenance à l'aide d'AWS CLI (p. 1409)	Exemples de nouvelles rubriques : L'enregistrement des cibles avec une fenêtre de maintenance fournit trois exemples de commandes pour illustrer les différentes façons de spécifier les cibles pour une fenêtre de maintenance lorsque vous utilisez l' AWS CLI . La rubrique explique également les meilleurs cas d'utilisation pour chacun des exemples de commandes.	May 3, 2019

Mises à jour des rubriques de groupes de correctifs (p. 1409)	La rubrique À propos des groupes de correctifs a été mise à jour pour inclure une section sur la façon dont les instances gérées déterminent la référence de correctifs appropriée lors des opérations d'application de correctifs. En outre, des instructions ont été ajoutées à l'aide de la AWS CLI console ou Systems Manager pour ajouter des balises Groupes de correctifs à vos instances gérées et expliquer comment ajouter un Groupe de correctifs pour une référence de correctifs. Pour de plus amples informations, veuillez consulter Créer un Groupe de correctifs et Ajouter un Groupe de correctifs pour une référence de correctifs.	May 1, 2019

Nouvelles fonctions Parameter Store (p. 1409)	Parameter Store propose les nouvelles fonctions suivantes : • Paramètres avancés : Parameter Store vous permet désormais de configurer individuellement les paramètres à utiliser un paramètre de niveau standard (valeur par défaut) ou un niveau d'utilisation entre les paramètres avancés. Les paramètres avancés offrent une plus grande taille limite pour la valeur du paramètre, une limite supérieure pour le nombre de paramètres que vous pouvez créer par compte et par région, et la possibilité d'utiliser des politiques de paramètre. Pour de plus amples informations sur les paramètres avancés, veuillez consulter la section À propos des paramètres avancés Systems Manager. • Stratégies de paramètre : Les stratégies de paramètre vous aident à gérer un ensemble croissant de paramètres en vous permettant d'attribuer des critères spécifiques à un paramètre, tels qu'une date d'expiration ou une durée de vie. Les stratégies de paramètre sont particulièrement utiles pour vous obliger à mettre à jour ou supprimer les mots de passe et les données de configuration stockés dans Parameter Store. Les stratégies de paramètre ne sont disponibles que pour les paramètres qui utilisent le niveau entre les paramètres avancés. Pour de plus amples informations, veuillez consulter Utilisation des stratégies de paramètres. • Plus haut débit : Vous pouvez désormais augmenter la limite Parameter Store de débit jusqu'à un maximum de 1 000 transactions par seconde. Pour de plus amples informations, veuillez consulter	April 25, 2019

	la section Accroissement du débit Parameter Store.	
Mise à jour de la section relative à Automation (p. 1409)	La section relative à Automation a été mise à jour pour améliorer la visibilité. En outre, quatre nouvelles rubriques ont été ajoutées à la section Automatisation : • Exécution manuelle d'une automatisation (p. 318) • Exécution d'une automatisation avec des approbateurs (p. 324) • Exécution d'automatisations basées sur des déclencheurs (p. 343)	April 17, 2019
Chiffrement des données de session à l'aide d'une clé AWS KMS (p. 1409)	Par défaut, Session Manager utilise TLS 1.2 pour chiffrer les données de session transmises entre les machines locales des utilisateurs de votre compte et vos instances EC2. Maintenant, vous pouvez choisir pour chiffrer ces données à l'aide d'une clé principale client (CMK) qui a été créée dans AWS Key Management Service. Vous pouvez utiliser une clé qui a été créée dans votre compte AWS ou une clé qui a été partagé avec vous à partir d'un autre compte. Pour de plus amples informations sur la spécification d'une CMK pour chiffrer les données de session, veuillez consulter Activer la clé de chiffrement AWS KMS des données de session (console), Créer des préférences Session Manager (AWS CLI) ou Mettre à jour les préférences Session Manager (AWS CLI).	April 4, 2019

Configuration de notifications Amazon SNS pour AWS Systems Manager (p. 1409)	Ajout des instructions d'utilisation de la console AWS CLI ou Systems Manager pour configurer les notifications Amazon SNS pour les tâches Fonctionnalité Exécuter la commande et Fonctionnalité Exécuter la commande enregistrées auprès d'une fenêtre de maintenance. Pour de plus amples informations, veuillez consulter Configuration des notifications Amazon SNS pour AWS Systems Manager.	March 6, 2019
Instances avancées pour les serveurs et VMs les environnements hybrides (p. 1409)	AWS Systems Managerpropose un niveau d'instances standard et un niveau d'instances avancées pour les serveurs et VMs dans votre environnement hybride. Le niveau d'instances standard vous permet d'enregistrer 1 000 serveurs au maximum ou VMs par compte AWS et par région AWS. Si vous devez enregistrer plus de 1 000 serveurs ou VMs dans un seul compte et une seule région, utilisez le niveau d'instances avancées. Vous pouvez créer autant d'instances que vous le souhaitez dans le niveau des instances avancées, mais toutes les instances configurées pour Systems Manager sont disponibles sur une base de paiement par utilisation. Les instances avancées vous permettent également de vous connecter à vos machines hybrides en utilisant AWS Systems Manager Session Manager. Session Manager fournit l'accès au shell interactif à vos instances. Pour de plus amples informations sur l'activation des instances avancées, veuillez consulter Utilisation du niveau des instances avancées.	March 4, 2019

Créer des associations State Manager qui utilisent des documents partagés SSM (p. 1409)	Vous pouvez créer des associations State Manager qui utilisent les documents Commande et Automation SSM partagés à partir d'autres comptes AWS. La création d'associations à l'aide de documents partagés SSM permet de conserver votre Amazon EC2 et l'infrastructure hybride dans un état cohérent, même lorsque les instances ne sont pas dans le même compte. Pour plus d'informations sur le partage des documents SSM, consultez AWS Systems Manager Documents. Pour de plus amples informations sur la création d'une association State Manager, veuillez consulter Créer une association.	February 28, 2019
Afficher des listes d'événements Systems Manager pris en charge pour les règles Amazon EventBridge (p. 1409)	La nouvelle rubrique Surveillance des événements Systems Manager avec Amazon EventBridge fournit un résumé des différents événements émis par Systems Manager pour lesquels vous pouvez configurer des règles de surveillance des événements dans EventBridge.	February 25, 2019
Ajouter des balises lorsque vous créez des ressources Systems Manager (p. 1409)	Systems Manager prend désormais en charge la possibilité d'ajouter des balises à certains types de ressources lorsque vous les créez. Les ressources que vous pouvez baliser lorsque vous les créez avec AWS CLI ou un kit SDK incluent des fenêtres de maintenance, des références de correctifs, des paramètres Parameter Store et des documents SSM. Vous pouvez également affecter des balises à une instance gérée lorsque vous créez une activation pour celle-ci. Lorsque vous utilisez la console Systems Manager, vous pouvez ajouter des balises à des fenêtres de maintenance, des références de correctifs et des paramètres.	February 24, 2019

Création automatique d'un rôle IAM pour l'inventaire Systems Manager (p. 1409)	Auparavant, vous deviez créer un rôle AWS Identity and Access Management (IAM) et attacher des stratégies distinctes à ce rôle pour afficher les données d'inventaire sur la page Affichage du détail d'inventaire dans la console. Vous n'avez plus besoin de créer ce rôle ni d'attacher des stratégies à celui-ci. Lorsque vous choisissez une synchronisation des données à distance sur la page Inventory Detail View, crée Systems Manager automatiquement le rôle AmazonGlueServicePolicyForSSM et lui attribue la stratégie AmazonGlueServicePolicyForSSM-{S3 bucket name} et la AWSGlueServiceRole stratégie correspondante. Pour de plus amples informations, veuillez consulter Interrogation des données d'inventaire à partir de plusieurs régions et comptes.	February 14, 2019
Procédures Fenêtres de maintenance pour mettre à jour Agent SSM (p. 1409)	Ajout de deux nouvelles procédures à la documentation Fenêtres de maintenance. Ces procédures décrivent l'utilisation de la console Systems Manager ou d'AWS CLI pour créer une fenêtre de maintenance qui maintient Agent SSM à jour automatiquement. Pour de plus amples informations, veuillez consulter Procédures Fenêtres de maintenance.	February 11, 2019
Utilisation de paramètres Parameter Store publics (p. 1409)	Ajout d'une brève section décrivant les paramètres Parameter Store publics. Pour de plus amples informations, veuillez consulter Utilisation des paramètres publics Systems Manager.	January 31, 2019

Utiliser AWS CLI pour créer des préférences Session Manager (p. 1409)	Ajout des instructions détaillées pour l'utilisation de l'AWS CLI afin de créer des préférences Session Manager, telles que les journaux CloudWatch, les options de journalisation de compartiment S3 et les paramètres de chiffrement de session. Pour de plus amples informations, veuillez consulter Utiliser l'AWS CLI pour créer des préférences Session Manager.	January 22, 2019
Exécution de flux de travail Systems Manager Automation avec State Manager (p. 1409)	AWS Systems Manager State Manager prend désormais en charge la création des associations qui utilisent des documents Automation SSM. Auparavant, State Manager prenait uniquement en charge les documents `command` et `policy`, ce qui signifie que vous pouviez uniquement créer des associations qui ciblaient des instances gérées. Grâce à la prise en charge des documents SSM Automation, vous pouvez désormais créer des associations qui ciblent les différents types de ressources AWS. Pour de plus amples informations, veuillez consulter les sections Exécution des flux de travail Systems Manager Automation avec State Manager.	January 22, 2019
Mises à jour des références pour les expressions Cron et Rate et pour les options de planification de fenêtre de maintenance (p. 1409)	La rubrique de référence Expressions cron et rate pour Systems Manager a été révisée. La nouvelle version fournit plus d'exemples et de meilleures explications sur la façon d'utiliser les expressions cron et rate pour planifier vos fenêtres de maintenance et les associations State Manager. En outre, la nouvelle rubrique Options de planification et de périodes actives Fenêtres de maintenance explique comment les différentes options liées à la planification des fenêtres de maintenance (date de début, date de fin, fuseau horaire, fréquence de planification) sont reliées entre elles.	December 6, 2018

Mises à jour de la rubrique Prérequis Systems Manager (p. 1409)	La rubrique Prérequis Systems Manager a été mise à jour afin de fournir des informations sur les versions de systèmes d'exploitation prises en charge dans un format tabulaire plus détaillé, avec d'autres modifications dans la page pour une meilleure lisibilité.	December 4, 2018
Activer la journalisation de débogage pour l'Agent SSM (p. 1409)	Vous pouvez activer la journalisation de débogage de l'Agent SSM en modifiant le fichier seelog.xml .template sur l'instance gérée. Pour de plus amples informations, veuillez consulter Activer la journalisation du débogage de l'Agent SSM.	November 30, 2018
Prise en charge des architectures de processeur ARM64 (p. 1409)	AWS Systems Manager prend désormais en charge les versions ARM64 des systèmes d'exploitation (18.04 LTS et 16.04 LTS) Amazon Linux 2, Red Hat Enterprise Linux 7.6, et Ubuntu Server. Pour plus d'informations, consultez les instructions d'installation Amazon Linux 2, RHEL, et Ubuntu Server 18.04 et 16.04 LTS avec les packages Snap. Pour de plus amples informations sur le type d'instance A1, veuillez consulter Instances à usage général dans le Amazon EC2 Guide de l'utilisateur pour les instances Linux.	November 26, 2018

Créez et déployez les packages en utilisant AWS Systems Manager Distributor (p. 1409)	vous AWS Systems Manager Distributor permet d'empaqueter vos propres logiciels ou de trouver des packages logiciels d'agent fournis par AWS , tels que AmazonCloudWatchAgent - pour installer sur des instances AWS Systems Manager gérées. Distributor publie des ressources, telles que des packages logiciels, sur des instances AWS Systems Manager gérées. La publication d'un package publie des versions spécifiques du document du package (un Systems Manager document que vous créez lorsque vous ajoutez le package dans Distributor) sur les instances gérées que vous identifiez par une instance gérée, IDs un compte AWS , des balises ou une IDs région .AWS Pour plus d'informations, consultez AWS Systems ManagerDistributor.	November 20, 2018
Exécution simultanée de flux de travail AWS Systems Manager Automation dans plusieurs régions AWS et comptes AWS à partir d'un compte principal (p. 1409)	Vous pouvez exécuter des flux de travail AWS Systems Manager Automation simultanément dans plusieurs régions AWS et comptes AWS ou unités d'organisation (UO) AWS à partir d'un compte de gestion Automation. Les automatisations exécutées simultanément dans plusieurs régions et comptes ou OUs réduisent le temps nécessaire pour administrer vos ressources AWS, tout en renforçant la sécurité de votre environnement informatique. Pour de plus amples informations, veuillez consulter Exécution des flux de travail Automation dans plusieurs régions et comptes AWS.	November 19, 2018

Interrogation de données d'inventaire depuis plusieurs régions et comptes AWS (p. 1409)	L'inventaire Systems Manager s'intègre à Amazon Athena pour vous aider à interroger les données d'inventaire provenant de plusieurs régions et comptes AWS. L'intégration d'Athena utilise la synchronisation de données de ressources pour que vous puissiez consulter les données d'inventaire depuis toutes vos instances gérées sur la page Inventory Detail View (Vue des détails d'inventaire), dans la console AWS Systems Manager. Pour de plus amples informations, veuillez consulter Interrogation des données d'inventaire à partir de plusieurs régions et comptes.	November 15, 2018
Création d'associations State Manager qui exécutent des fichiers MOF (p. 1409)	Vous pouvez exécuter des fichiers MOF (Managed Object Format) pour appliquer un état souhaité aux instances gérées de Windows Server avec State Manager en utilisant le document AWS-ApplyDSCMofs SSM. Le document AWS-ApplyDSCMofs a deux modes d'exécution. Avec le premier mode, vous pouvez configurer l'association pour analyser et indiquer si les instances gérées sont actuellement dans l'état souhaité défini dans les fichiers MOF spécifiés. Dans le second mode, vous pouvez exécuter les fichiers MOF et modifier la configuration de vos instances basées sur les ressources et leurs valeurs définies dans les fichiers MOF. Le document AWS-ApplyDSCMofs vous permet de télécharger et d'exécuter les fichiers de configuration MOF à partir de Amazon Simple Storage Service (Amazon S3), d'un partage local, ou à partir d'un site Web sécurisé avec un domaine HTTPS. Pour de plus amples informations, veuillez consulter la section Création d'associations qui exécutent des fichiers MOF.	November 15, 2018

Restreindre l'accès administratif dans les sessions Session Manager (p. 1409)	Les Session Manager sessions sont lancées à l'aide des informations d'identification d'un compte utilisateur créé avec les privilèges racine ou administrateur par défaut appelés `ssm-user` . Des informations sur la restriction du contrôle administratif pour ce compte sont désormais disponibles dans la rubrique Désactiver ou activer les autorisations administratives de compte ssm-user.	November 13, 2018
Exemples YAML dans Référence d'actions Automation (p. 1409)	La référence d'actions d'automatisation inclut désormais un modèle YAML pour chaque action qui comprend déjà un modèle JSON.	October 31, 2018
Attribuer des niveaux de gravité de conformité aux associations (p. 1409)	Vous pouvez désormais attribuer des niveaux de gravité de conformité pour les associations State Manager. Ces niveaux de gravité sont présentés dans le tableau de bord de conformité et peuvent également être utilisés pour filtrer vos rapports de conformité. Les niveaux de gravité que vous pouvez attribuer incluent : Critique, Élevé, Moyen, Faible et Non précisé. Pour de plus amples informations, veuillez consulter Créer une association (console).	October 26, 2018
Utilisation de cibles et de contrôles du débit avec Automation et State Manager (p. 1409)	Contrôlez l'exécution des automatisations et des associations State Manager dans l'ensemble de votre parc de ressources en utilisant des cibles, la simultanéité et les seuils d'erreur. Pour de plus amples informations, veuillez consulter Utilisation des cibles et des contrôles de débit pour l'exécution des flux de travail Automation d'une flotte et Utilisation des cibles et des contrôles de débit avec les associations State Manager.	October 23, 2018

Spécification de plages de temps actives et de fuseaux horaires internationaux pour les fenêtres de maintenance (p. 1409)	Vous pouvez également spécifier des dates avant lesquelles ou après lesquelles (date de début et date de fin) une fenêtre de maintenance ne devrait pas s'exécuter et vous pouvez spécifier le fuseau horaire international sur lequel baser le calendrier de la fenêtre de maintenance. Pour de plus amples informations, veuillez consulter Création d'une fenêtre de maintenance (console) et Mise à jour d'une fenêtre de maintenance (AWS CLI).	October 9, 2018
Gestion d'une liste personnalisée de correctifs pour votre référence de correctifs dans un compartiment S3 (p. 1409)	Le nouveau paramètre 'InstallOverrideList' dans le document SSM 'AWS-RunPatchBaseline vous permet de spécifier une URL https ou une URL chemin d'accès Amazon Simple Storage Service (Amazon S3) pour obtenir la liste des correctifs à installer. Cette liste d'installation de correctifs que vous gérez dans un compartiment S3 au format YAML remplace les correctifs spécifiés par la référence de correctifs par défaut. Pour plus d'informations, consultez Nom du paramètre : InstallOverrideList.	October 5, 2018
Contrôle étendu si les dépendances de correctifs sont installées (p. 1409)	Auparavant, si un correctif de votre liste des correctifs rejetés était identifié comme dépendant d'un autre correctif, il était quand même installé. Maintenant, vous pouvez choisir s'il convient d'installer ces dépendances ou de bloquer leur installation. Pour de plus amples informations, veuillez consulter Créer une référence de correctif.	October 5, 2018

Créer des flux de travail Automation dynamiques avec des ramifications conditionnelles (p. 1409)	L'action Automation `aws:branch` vous permet de créer un flux de travail Automation dynamique qui évalue plusieurs options en une seule étape, puis passe à une autre étape dans le document Automation en fonction des résultats de cette évaluation. Pour de plus amples informations, veuillez consulter Création de flux de travail Automation dynamiques avec branchement conditionnel.	September 26, 2018
Utiliser l'AWS CLI pour mettre à jour les préférences Session Manager (p. 1409)	Des instructions pour l'utilisation de l'interface de ligne de commande pour mettre à jour les préférences Session Manager, telles que les journaux CloudWatch et les options de journalisation des compartiments S3, ont été ajoutées au Guide de l'utilisateur AWS Systems Manager. Pour de plus amples informations, veuillez consulter Utilisation de l'AWS CLI pour la mise à jour des préférences Session Manager.	September 25, 2018
Configurez plus facilement les options d'application des correctifs avec la nouvelle page « Configuration de l'application des correctifs » (p. 1409)	Gestionnaire de correctifsa été mis à jour avec un nouveau système pour configurer l'application des correctifs. Sur la page Configure patching (Configurer l'application de correctifs), vous pouvez spécifier plusieurs options pour l'application des correctifs dans un emplacement unique, notamment en associant une fenêtre de maintenance à une configuration d'application de correctif et en modifiant la référence de correctifs associée à un groupe de correctifs. Pour de plus amples informations, veuillez consulter À propos des configurations d'application de correctifs et Création d'une configuration d'application de correctifs.	September 22, 2018

Exigence de Agent SSM mise à jour pour Session Manager (p. 1409)	Session Manager exige désormais Agent SSM version 2.3.68.0 ou versions ultérieures. Pour de plus amples informations sur les prérequis de Session Manager, veuillez consulter Exécuter les opérations Session Manager prérequises.	September 17, 2018
Gestion des instances sans ouverture des ports entrants ou maintenance des hôtes bastion utilisant Session Manager (p. 1409)	Maintenant disponible, Session Manager est une fonctionnalité AWS Systems Manager entièrement gérée qui vous permet de gérer vos instances EC2 en un clic via un navigateur interactif Shell ou via l'AWS CLI. Session Manager fournit une gestion des instances sûre et vérifiable sans qu'il soit nécessaire d'ouvrir les ports entrants, de maintenir les hôtes bastion ou de gérer des clés SSH. Session Manager facilite également la mise en conformité avec les politiques d'entreprise qui nécessitent un accès contrôlé aux instances, des pratiques de sécurité strictes, et des journaux totalement vérifiables avec les détails d'accès aux instances, tout en fournissant aux utilisateurs finaux l'accès inter plateforme à vos instances EC2 d'un simple clic. Pour de plus amples informations, veuillez consulter Pour en savoir plus sur Session Manager.	September 11, 2018
Appel d'autres services AWS à partir d'un flux de travail Systems Manager Automation (p. 1409)	Vous pouvez appeler d'autres services AWS et d'autres fonctionnalités Systems Manager dans votre flux de travail Automation à l'aide de trois nouvelles actions (ou plug-ins) Automation dans vos documents Automation. Pour de plus amples informations, veuillez consulter Appel d'autres services AWS à partir d'un flux de travail Systems Manager Automation.	August 28, 2018

Utilisation de clés de condition spécifiques - Systems Manager dans les stratégies IAM (p. 1409)	La rubrique Spécification des conditions dans une stratégie a été mise à jour pour répertorier les clés de condition IAM pour Systems Manager que vous pouvez incorporer dans les stratégies. Vous pouvez utiliser ces clés pour spécifier les conditions dans lesquelles une stratégie doit prendre effet. La rubrique inclut également des liens vers des exemples de stratégies et d'autres rubriques connexes.	August 18, 2018
Agrégation des données d'inventaire avec des groupes pour voir quelles instances sont ou non configurées pour collecter un type d'inventaire (p. 1409)	Les groupes vous permettent de voir rapidement un décompte des instances gérées qui sont ou ne sont pas configurées pour collecter un ou plusieurs types d'inventaire. Avec les groupes, vous spécifiez un ou plusieurs types d'inventaire et un filtre qui utilise l'opérateur `exists`. Pour de plus amples informations, veuillez consulter Agrégation des données d'inventaire.	August 16, 2018
Affichage du suivi des modifications et de l'historique pour l'inventaire et la conformité de configuration (p. 1409)	Vous pouvez désormais afficher le suivi des modifications et l'historique pour l'inventaire collecté à partir de vos instances gérées. Vous pouvez également afficher le suivi des modifications et l'historique pour les Gestionnaire de correctifs correctifs et State Manager associations signalés par le service Conformité de configuration. Pour de plus amples informations, veuillez consulter Affichage du suivi des modifications et de l'historique d'inventaire.	August 9, 2018

Le rôle lié au service Systems Manager étend la prise en charge des tâches de fenêtre de maintenance (p. 1409)	Le service Fenêtres de maintenance exige un ensemble d'autorisations IAM pour pouvoir exécuter les tâches de fenêtre de maintenance sur vos instances. Auparavant, la seule option consistait à créer un rôle IAM personnalisé pour fournir ces autorisations. Le rôle lié à un service pour Systems Manager a désormais été amélioré pour fournir ces autorisations, ce qui vous offre deux options de rôle IAM. Pour de plus amples informations, veuillez consulter Dois-je utiliser un rôle lié à un service ou un rôle de service personnalisé pour exécuter des tâches de fenêtre de maintenance ?	August 2, 2018
Parameter Store s'intègre à Secrets Manager. (p. 1409)	Parameter Store est désormais intégré à AWS Secrets Manager afin que vous puissiez récupérer les secrets Secrets Manager lors de l'utilisation d'autres services AWS qui prennent déjà en charge les références aux paramètres Parameter Store. Ces services incluent Amazon EC2, Amazon Elastic Container Service, AWS Lambda, AWS CloudFormation, AWS CodeBuild, AWS CodeDeploy, et d'autres fonctionnalités Systems Manager. Si vous utilisez Parameter Store pour référencer les secrets Secrets Manager, vous créez un processus cohérent et sécurisé permettant d'appeler et d'utiliser les secrets, ainsi que de référencer les données dans votre code et vos scripts de configuration. Pour de plus amples informations, veuillez consulter Référencement des secrets AWS Secrets Manager à partir des paramètres Parameter Store.	July 26, 2018

Attacher des étiquettes aux paramètres Parameter Store (p. 1409)	Une étiquette de paramètre est un alias défini par l'utilisateur pour vous aider à gérer les différentes versions d'un paramètre. Lorsque vous modifiez un paramètre, Systems Manager enregistre automatiquement une nouvelle version et incrémente le numéro de version d'une unité. Une étiquette peut vous aider à vous souvenir de l'objectif d'une version de paramètre lorsqu'il existe plusieurs versions. Pour de plus amples informations, veuillez consulter Étiquetage des paramètres.	July 26, 2018
Création de flux de travail Automation dynamiques (p. 1409)	Par défaut, les étapes (ou actions) que vous définissez dans la mainSteps section d'un document Automation sont exécutées par ordre séquentiel. Lorsqu'une action est terminée, la prochaine action spécifiée dans la mainSteps section commence. Avec cette version, vous pouvez désormais créer des flux de travail Automation qui effectuent des ramifications conditionnelles. Ainsi, vous pouvez créer des flux de travail Automation qui répondent de manière dynamique aux changements de conditions et passent à une étape spécifiée. Pour de plus amples informations, veuillez consulter Création de flux de travail Automation dynamiques.	July 18, 2018
Agent SSMdésormais préinstallé sur Ubuntu Server 16.04 AMIs à l'aide de Snap (p. 1409)	Depuis les instances créées à partir de Ubuntu Server 16.04 AMIs identifiées par `20180627` , l' Agent SSM est préinstallé à l'aide de packages Snap. Pour les instances créées à partir d' AMIs , vous devez continuer à utiliser les packages d'installation deb. Pour plus d'informations, consultez A propos des installations de l'Agent SSM sur les instances 16.04 64 bits Ubuntu Server.	July 7, 2018

Vérification des autorisations S3 minimales requises par Agent SSM (p. 1409)	La nouvelle rubrique Autorisations minimales relatives au compartiment S3 pour l'Agent SSM fournit des informations sur les compartiments Amazon Simple Storage Service (Amazon S3) auxquels les ressources peuvent avoir besoin d'accéder pour effectuer des opérations Systems Manager. Vous pouvez spécifier ces compartiments dans une stratégie personnalisée si vous souhaitez limiter l'accès au compartiment S3 pour un profil d'instance ou le point de terminaison d'un VPC au minimum requis pour utiliser Systems Manager.	July 5, 2018
Affichage de l'historique d'exécution complet correspondant à un ID d'association State Manager spécifique (p. 1409)	La nouvelle rubrique Affichage des historiques des associations décrit comment afficher toutes les exécutions correspondant à un ID d'association spécifique, puis afficher les détails d'exécution d'une ou de plusieurs ressources.	July 2, 2018
Gestionnaire de correctifsintroduit la prise en charge d'Amazon Linux 2 (p. 1409)	Vous pouvez désormais utiliser Gestionnaire de correctifs pour appliquer des correctifs aux Amazon Linux 2 instances. Pour obtenir des informations générales sur la prise en charge du système Gestionnaire de correctifs d'exploitation, consultez les Gestionnaire de correctifs conditions préalables. Pour plus d'informations sur les paires clé-valeur prises en charge pour Amazon Linux 2 lors de la définition d'un filtre de correctifs, consultez PatchFilter dans le Référence d'API AWS Systems Manager .	June 26, 2018
Envoi de la sortie de commande à Amazon CloudWatch Logs (p. 1409)	La nouvelle rubrique Configuration de Amazon CloudWatch Logs pour Fonctionnalité Exécuter la commande décrit comment envoyer la sortie de Fonctionnalité Exécuter la commande vers CloudWatch Logs.	June 18, 2018

Créez ou supprimez rapidement la synchronisation des données de ressource pour l'inventaire en utilisant AWS CloudFormation (p. 1409)	Vous pouvez également utiliser AWS CloudFormation pour créer ou supprimer une synchronisation des données de ressources pour l'inventaire Systems Manager. Pour utiliser AWS CloudFormation, ajoutez la ressource AWS::SSM::ResourceDataSync à votre modèle AWS CloudFormation. Pour plus d'informations, consultez Travail avec les modèles AWS CloudFormation dans le AWS CloudFormation Guide de l'utilisateur. Vous pouvez également créer manuellement une synchronisation des données de ressource pour l'inventaire comme décrit dans Configuration de la synchronisation des données de ressource pour l'inventaire.	June 11, 2018
Notifications de mise à jour du Guide de l'utilisateur AWS Systems Manager désormais disponibles via RSS (p. 1409)	La version HTML du Guide de l'utilisateur Systems Manager prend désormais en charge un flux RSS des mises à jour qui sont documentées sur la page Historique des mises à jour de la documentation Systems Manager. Le flux RSS inclut les mises à jour effectuées à partir de juin 2018. Les mises à jour annoncées précédemment sont toujours disponibles sur la page Historique de mise à jour de la documentation Systems Manager. Utilisez le bouton RSS dans le panneau du menu supérieur pour vous abonner au flux.	June 6, 2018
Spécifiez un code de sortie dans les scripts afin de redémarrer les instances gérées (p. 1409)	La nouvelle rubrique Redémarrage d'une Instance gérée à partir de scripts décrit comment charger Systems Manager de redémarrer les instances gérées en spécifiant un code de sortie dans les scripts que vous exécutez avec Fonctionnalité Exécuter la commande.	June 3, 2018

Créer un événement dans Amazon EventBridge chaque fois que l'inventaire personnalisé est supprimé (p. 1409)	La nouvelle rubrique Affichage des actions de suppression de l'inventaire dans EventBridge décrit comment configurer Amazon EventBridge pour créer un événement à chaque fois qu'un utilisateur supprime l'inventaire personnalisé.	June 1, 2018

Mises à jour antérieures

Le tableau ci-après décrit les modifications importantes apportées à chaque publication du Guide de l'utilisateur AWS Systems Manager avant le 27 juin 2018.

Modification	Description	Date de publication
Effectuer l'inventaire de toutes les instances gérées dans votre compte AWS	Vous pouvez facilement inventorier toutes les instances gérées dans votre compte AWS en créant une association d'inventaire global. Pour plus d'informations, consultez la section Effectuer l'inventaire de toutes les instances gérées dans votre compte AWS (p. 807). Note Les associations d'inventaire global sont disponibles dans Agent SSM version 2.0.790.0 ou ultérieure. Pour plus d'informations sur la mise à jour de l'Agent SSM sur vos instances, consultez Mettre à jour l'Agent SSM à l'aide de Fonctionnalité Exécuter la commande (p. 948).	3 mai 2018
Agent SSM installé par défaut sur Ubuntu Server 18	Agent SSMest installé par défaut sur Ubuntu Server 18.04 LTS 64 bits et 32 bitsAMIs.	2 mai 2018
Nouvelle rubrique	La nouvelle rubrique Exécution de commandes à l'aide du paramètre de version de document (p. 953) décrit comment utiliser le paramètre version-document pour spécifier la version d'un document SSM à utiliser lors de l'exécution de la commande.	1 mai 2018
Nouvelle rubrique	La nouvelle rubrique Suppression de l'inventaire personnalisé (p. 827) décrit comment supprimer des données d'inventaire personnalisées à partir de Amazon S3 à l'aide de l'AWS CLI. Cette rubrique explique également comment utiliser l'option `SchemaDeleteOption` pour gérer l'inventaire personnalisé en désactivant ou en supprimant un type d'inventaire personnalisé. Cette nouvelle fonction utilise l'action DeleteInventory d'API .	19 avril 2018
Notifications Amazon SNS pour l'Agent SSM	Vous pouvez vous abonner à une rubrique Amazon SNS pour recevoir des notifications lorsqu'une nouvelle version de l'Agent SSM est disponible. Pour plus d'informations, consultez la section Abonnement aux notifications Agent SSM (p. 111).	9 avril 2018

Modification	Description	Date de publication
Prise en charge de CentOS l'application de correctifs	prend Systems Manager désormais en charge les CentOS instances d'application de correctifs. Pour plus d'informations sur CentOS les versions prises en charge, consultez Prérequis d'Gestionnaire de correctifs (p. 1048) . Pour en savoir plus sur la façon dont les correctifs fonctionnent, consultez Fonctionnement Gestionnaire de correctifs des opérations (p. 1049).	29 mars 2018
Nouvelle section	Pour fournir une seule source pour les informations de référence dans le Guide de l'utilisateur AWS Systems Manager, une nouvelle section a été ajoutée, Référence AWS Systems Manager (p. 1387). Du contenu supplémentaire sera ajouté à cette section dès qu'il sera disponible.	15 mars 2018
Nouvelle rubrique	La nouvelle rubrique À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés (p. 1103) détaille les formats de noms de package que vous pouvez saisir dans les listes de correctifs approuvés et rejetés pour une référence de correctifs personnalisée. Des exemples de formats sont fournis pour chaque type de système d'exploitation pris en charge par Gestionnaire de correctifs .	9 mars 2018
Nouvelle rubrique	Systems Managers'intègre désormais à InSpec Chef InSpec . est une infrastructure d'exécution open source qui vous permet de créer des profils lisibles par l'utilisateur sur GitHub ou Amazon S3 . Ensuite, vous pouvez utiliser Systems Manager pour exécuter des analyses de conformité et afficher les instances conformes et non conformes. Pour plus d'informations, consultez la section Utilisation des InSpec profils Chef avec Systems Manager la conformité (p. 145).	7 mars 2018
Nouvelle rubrique	La nouvelle rubrique Utilisation des rôles liés à un service pour Systems Manager (p. 1312) indique comment utiliser un rôle lié à un service AWS Identity and Access Management (IAM) avec Systems Manager. À l'heure actuelle, les rôles liés à un service sont uniquement requis lorsque vous utilisez l'inventaire Systems Manager pour collecter les métadonnées sur les balises et les groupes de ressources.	27 février 2018

Modification	Description	Date de publication
Rubriques nouvelles et mises à jour	Vous pouvez désormais utiliser Gestionnaire de correctifs pour installer des correctifs qui se trouvent dans un référentiel source différent de celui par défaut configuré sur l'instance. Cela s'avère utile pour appliquer des correctifs à des instances via des mises à jour non liées à la sécurité, le contenu des dépôts PPA (Personal Package Archive) pour Ubuntu Server, des mises à jour destinées à des applications d'entreprise internes, et ainsi de suite. Vous pouvez spécifier d'autres référentiels source de correctifs lors de la création d'une référence de correctifs personnalisée. Pour de plus amples informations, consultez les rubriques suivantes : • Spécification d'un autre référentiel source de correctifs (Linux) (p. 1054) • Utilisation de références de correctifs personnalisées (p. 1127) • Création d'une référence de correctifs avec des référentiels personnalisés pour les différentes versions du système d'exploitation (p. 1149) En outre, vous pouvez désormais utiliser Gestionnaire de correctifs pour corriger SUSE Linux Enterprise Server les instances . Gestionnaire de correctifs prend en charge l'application de correctifs aux versions SLES 12.* (64 bits uniquement). Pour en savoir plus, consultez les informations spécifiques à SLES dans les rubriques suivantes : • Sélection des correctifs de sécurité (p. 1049) • Installation des correctifs (p. 1056) • Fonctionnement des règles de référence de correctif sur SUSE Linux Enterprise Server (p. 1073)	6 février 2018
Nouvelle rubrique	La nouvelle rubrique Mettre à niveau le module de demandes Python sur Amazon Linux les instances qui utilisent un serveur proxy (p. 98) fournit des instructions pour s'assurer que les instances créées à l'aide d'une Amazon Linux AMI ont été mises à jour avec une version actuelle de l' Python `requests`du module. Cette exigence vise à garantir la compatibilité avec Gestionnaire de correctifs .	12 janvier 2018
Nouvelle rubrique	La nouvelle rubrique À propos des documents SSM pour les instances d'application de correctifs (p. 1076) décrit les sept documents SSM actuellement disponibles pour vous aider à appliquer les dernières mises à jour de sécurité à vos instances gérées.	10 janvier 2018

Modification	Description	Date de publication
Mises à jour importantes concernant la prise en charge de Linux	Mise à jour de différentes rubriques avec les informations suivantes : • Agent SSMest installé par défaut sur Amazon Linux une base AMIs de base datée de septembre 2017 et après. • Vous devez installer manuellement Agent SSM sur les autres versions de Linux, y compris les images ne faisant pas partie de la base de données, comme les images optimisées pour Amazon ECSAMIs.	9 janvier 2018
Nouvelle rubrique	Une nouvelle rubrique, À propos du Document SSM AWS-RunPatchBaseline (p. 1080), explique le fonctionnement de ce document SSM sur les systèmes Windows et Linux. Elle fournit également des informations sur les deux paramètres disponibles dans le document AWS-RunPatchBaseline. `Operation` et `Snapshot ID`.	5 janvier 2018
Nouvelles rubriques	Une nouvelle section, Fonctionnement Gestionnaire de correctifs des opérations (p. 1049) , fournit des détails techniques sur la façon dont Gestionnaire de correctifs détermine les correctifs de sécurité à installer et les installe sur chaque système d'exploitation pris en charge. Elle fournit également des informations sur le fonctionnement des règles de référence de correctif sur différentes distributions du système d'exploitation Linux.	2 janvier 2018
Référence aux actions Automation Systems Manager renommée et déplacée	En réaction aux commentaires des clients, la référence aux actions Automation porte désormais le nom de « document Automation Systems Manager de référence ». Celui-ci a été déplacé vers le nœud Ressources partagées > Documents pour être plus près de Référence du plug-in de document Systems Manager (p. 1232). Pour plus d'informations, consultez la section Référence sur les actions Systems Manager Automation (p. 386).	20 décembre 2017
Nouveau chapitre et contenu sur la surveillance	Un nouveau chapitre Surveillance des AWS Systems Manager (p. 1326) contient des instructions relatives à l'envoi de métriques et de données de journaux à Amazon CloudWatch Logs. Une nouvelle rubrique, Envoi de journaux d'instance à CloudWatch Logs (Agent CloudWatch) (p. 1327), contient des instructions relative à la migration de tâches de surveillance sur l'instance, sur des instances Windows Server 64 bits uniquement, de l'Agent SSM vers l'Agent CloudWatch.	14 décembre 2017
Nouveau chapitre	Un nouveau chapitre, Identity and Access Management pour AWS Systems Manager (p. 1293), fournit des informations complètes sur l'utilisation d'AWS Identity and Access Management (IAM) et d'AWS Systems Manager afin de sécuriser l'accès à vos ressources à l'aide d'informations d'identification. Ces informations d'identification octroient les autorisations requises pour accéder aux ressources AWS, notamment les données stockées dans des compartiments S3 et envoyer des commandes aux instances EC2 et y lire des balises.	11 décembre 2017

Modification	Description	Date de publication
Changements apportés à la navigation sur la gauche	Nous avons modifié les en-têtes dans la navigation sur la gauche de ce guide de l'utilisateur afin qu'ils correspondent aux en-têtes de la nouvelle console AWS Systems Manager.	8 décembre 2017
Diverses modifications pour re:Invent 2017	• Lancement officiel d'AWS Systems Manager : AWS Systems Manager (auparavant Amazon EC2 Systems Manager) est une interface unifiée qui vous permet de centraliser facilement des données opérationnelles et d'automatiser des tâches dans vos ressources AWS. Vous pouvez accéder à la nouvelle console AWS Systems Managerici. Pour plus d'informations, veuillez consulter Présentation du AWS Systems Manager (p. 1) • Prise en charge de YAML : Vous pouvez créer SSM des documents au format YAML. Pour plus d'informations, consultez la section Documents AWS Systems Manager (p. 1211).	29 novembre 2017
Utilisation de Fonctionnalité Exécuter la commande pour prendre des instantanés VSS de volumes EBS	À l'aide de Fonctionnalité Exécuter la commande, vous pouvez prendre des instantanés cohérents par rapport à l'application de tous les volumes Amazon Elastic Block Store (Amazon EBS) attachés à vos instances Windows Amazon EC2. Le processus d'instantané utilise le service Volume Shadow Copy Service (VSS) de Windows pour exécuter des sauvegardes au niveau des images des applications compatibles VSS, y compris des données d'opérations en cours entre ces applications et le disque. Par ailleurs, vous n'avez pas besoin de fermer vos instances ni de les déconnecter lorsque vous devez sauvegarder tous les volumes attachés. Pour plus d'informations, consultez Utilisation de la fonctionnalité Exécuter la commande pour prendre des instantanés VSS de volumes EBS dans le Amazon EC2 Guide de l'utilisateur pour les instances Windows.	20 novembre 2017
Mise à disposition de la sécurité Systems Manager améliorée à l'aide de points de terminaison VPC	Vous pouvez renforcer la sécurité de vos instances gérées (notamment de celles figurant dans votre environnement hybride) en configurant Systems Manager pour qu'il utilise un point de terminaison VPC d'interface. Les points de terminaison d'interface reposent sur PrivateLink , une technologie qui vous permet d'accéder de façon privée à Amazon EC2 et à l'aide d'adresses IP privées.Systems ManagerAPIs PrivateLinklimite tout le trafic réseau entre vos instances gérées, et EC2 au réseau Amazon (les instances gérées n'ont pas accès à Internet).Systems Manager De même, vous n'avez pas besoin d'une passerelle Internet, d'un périphérique NAT ou d'une passerelle privée virtuelle. Pour plus d'informations, consultez la section (Facultatif) Créer un point de terminaison de VPC (p. 42).	7 novembre 2017

Modification	Description	Date de publication
Prise en charge de l'inventaire pour les fichiers, les services, les rôles Windows et le registre Windows	SSM Inventory prend désormais en charge la collecte des informations suivantes à partir de vos instances gérées. • Fichiers : Nom, taille, version, date d'installation, heures de modification et des derniers accès, etc. • Services : Nom, nom d'affichage, statut, services dépendants, type de service, type de démarrage, etc. • Registre Windows : Chemin de la clé de registre, nom de valeur, type de valeur et valeur. • Rôles Windows : Nom, nom d'affichage, chemin, type de fonctionnalité, état installé, etc. Avant de tenter de collecter des informations pour ces types d'inventaire, mettez à jour Agent SSM sur les instances à inventorier. En exécutant la dernière version de Agent SSM, vous êtes sûr de collecter les métadonnées de tous les types d'inventaire pris en charge. Pour plus d'informations sur la mise à jour de l'Agent SSM à l'aide de State Manager, consultez Mise à jour automatique d'Agent SSM (interface de ligne de commande) (p. 1043). Pour en savoir plus sur Inventory, consultez En savoir plus sur l'inventaire Systems Manager (p. 794).	6 novembre 2017
Mises à jour de la documentation Automation	Plusieurs problèmes ont été résolus dans les informations relatives à la définition et la configuration de l'accès à Systems Manager Automation. Pour plus d'informations, consultez la section Configuration d'Automation (p. 308).	31 octobre 2017

Modification	Description	Date de publication
GitHubIntégration etAmazon S3	Exécuter des scripts distants : prend Systems Manager désormais en charge le téléchargement et l'exécution de scripts à partir d'un GitHub référentiel public ou privé, et à partir d' Amazon S3 . A l'aide du `AWS-RunRemoteScript` document SSM prédéfini ou du `aws:downloadContent` plug-in dans un SSM document personnalisé, vous pouvez exécuter des manuels stratégiques et des scripts Ansible dans Python, Ruby ou PowerShell , pour n'en citer que quelques-uns. Ces modifications améliorent encore l'infrastructure en tant que code lorsque vous utilisez Systems Manager pour automatiser la configuration et le déploiement d'instances EC2 et d'instances gérées sur site dans votre environnement hybride. Pour de plus amples informations, veuillez consulter Exécution de scripts depuis GitHub (p. 139) et Exécution de scripts depuis Amazon S3 (p. 126). Créer des documents SSM composites : Systems Manager prend désormais en charge l'exécution d'un ou de plusieurs documents SSM secondaires à partir d'un document SSM principal. Ces documents principaux qui exécutent d'autres documents sont appelés documents composites. Les documents composites permettent de créer et de partager un ensemble standard de documents SSM secondaires entre plusieurs comptes AWS, pour des tâches courantes telles que l'amorçage d'un logiciel antivirus ou la jonction d'instances à un domaine. Vous pouvez exécuter des documents composites et secondaires stockés dans Systems Manager , ou GitHub .Amazon S3 Après avoir créé un document composite, vous pouvez l'exécuter en utilisant le document SSM prédéfini `AWS-RunDocument`. Pour de plus amples informations, veuillez consulter Création de documents composites (p. 1273) et Exécution de documents Systems Manager de commande à partir d'emplacements distants (p. 1286). Référence SSM du plug-in de document : Pour faciliter l'accès, nous avons déplacé la Référence de SSM plug-in pour SSM les documents hors de la Référence dSystems Manager'API et dans le Guide de l'utilisateur . Pour plus d'informations, consultez la section Référence du plug-in de document Systems Manager (p. 1232).	26 octobre 2017

Modification	Description	Date de publication
Prise en charge des versions de paramètre dans Parameter Store	Lorsque vous modifiez un paramètre, Parameter Store itère désormais automatiquement le numéro de version de 1. Vous pouvez spécifier un nom et un numéro de version spécifiques pour un paramètre dans les appels d'API et les documents SSM. Si vous ne spécifiez pas de numéro de version, le système utilise automatiquement la dernière version. Les versions de paramètre fournissent une couche de protection au cas où un paramètre serait modifié par erreur. Vous pouvez afficher les valeurs de toutes les versions, et référencer des versions plus anciennes, si nécessaire. Vous pouvez également utiliser les versions de paramètre pour savoir combien de fois un paramètre a été modifié sur une période donnée. Pour plus d'informations, consultez la section Utilisation des versions de paramètre (p. 296).	24 octobre 2017
Prise en charge du balisage des documents Systems Manager	Vous pouvez désormais utiliser l'AddTagsToResourceAPI AWS CLI , l' ou l' Outils AWS pour PowerShell pour baliser Systems Manager les documents avec des paires clé-valeur. Le balisage vous aide à identifier rapidement des ressources spécifiques en fonction des balises que vous leur avez attribuées. Ceci s'ajoute à la prise en charge existante du balisage pour les instances gérées, les fenêtres de maintenance, les paramètres Parameter Store et les références de correctifs. Pour plus d'informations, consultez la section Balisage de documents Systems Manager (p. 1360).	3 octobre 2017
Mises à jour variées de la documentation pour corriger les erreurs et mettre à jour le contenu en fonction des commentaires	• Mise à jour de Configuration d'AWS Systems Manager pour les environnements hybrides (p. 49) avec des informations pour Raspbian Linux. • Mise à jour Configuration d'AWS Systems Manager (p. 28) avec les nouvelles exigences pour Windows Server les instances. Agent SSM nécessite Windows PowerShell 3.0 ou version ultérieure pour exécuter certains SSM documents sur Windows Server les instances (par exemple, le document SSM AWS-ApplyPatchBaseline). Vérifiez que vos instances Windows Server exécutent Windows Management Framework 3.0 ou version ultérieure. comprend PowerShell . Pour plus d'informations, consultez Windows Management Framework 3.0.	2 octobre 2017
Dépannage des instances Windows inaccessibles à l'aide du flux de travail EC2Rescue Automation	EC2Rescuepeut vous aider à diagnostiquer et à résoudre les problèmes sur Amazon EC2 Windows Server les instances . Vous pouvez exécuter l'outil en tant que flux de travail Systems Manager Automation à l'aide du document AWSSupport -ExecuteEC2Rescue. Le document AWSSupport -ExecuteEC2Rescue est conçu pour exécuter une combinaison d'Systems Manageractions , d'AWS CloudFormationactions et de Lambda fonctions qui automatisent les étapes normalement requises pour utiliser EC2Rescue . Pour plus d'informations, consultez Procédure : Exécuter l'EC2Rescueoutil sur des instances inaccessibles (p. 656) .	29 septembre 2017

Modification	Description	Date de publication
Agent SSM installé par défaut sur Amazon Linux	Agent SSML' est installé par défaut le Amazon Linux AMIs septembre 2017 et après. Vous devez installer manuellement l'Agent SSM sur les autres versions de Linux, comme décrit dans Installation et configuration de l'Agent SSM sur les instances EC2 pour Linux (p. 75).	27 septembre 2017
Améliorations de Fonctionnalité Exécuter la commande	Fonctionnalité Exécuter la commande inclut les améliorations suivantes. • Vous pouvez restreindre l'exécution de commandes à des instances spécifiques en créant une stratégie d'utilisateur IAM qui comporte une condition selon laquelle l'utilisateur ne peut exécuter de commandes que sur les instances comportant des balises Amazon EC2 spécifiques. Pour plus d'informations, consultez la section Restriction de l'accès de la fonctionnalité Fonctionnalité Exécuter la commande en fonction des balises d'instance (p. 943). • Vous disposez de plusieurs options pour cibler les instances à l'aide de balises Amazon EC2. Vous pouvez désormais spécifier plusieurs clés et valeurs de balise lors de l'envoi de commandes. Pour plus d'informations, consultez la section Utilisation des contrôles de cibles et de taux pour envoyer des commandes à un parc (p. 955).	12 septembre 2017
Systems Manager pris en charge sur Raspbian	Systems Manager peut désormais s'exécuter sur les appareils Raspbian Jessie et Raspbian Stretch, Raspberry Pi (32 bits) inclus.	7 septembre 2017
Envoyer automatiquement les journaux de l'Agent SSM à Amazon CloudWatch Logs	Vous pouvez maintenant effectuer une simple modification de configuration sur vos instances de telle sorte que l'Agent SSM envoie les fichiers journaux à CloudWatch. Pour plus d'informations, consultez la section Envoi de Agent SSM journaux àCloudWatch Logs (p. 1333).	7 septembre 2017
Chiffrer la synchronisation des données des ressources	La synchronisation des données de ressources Systems Manager vous permet de regrouper les données d'inventaire collectées sur des dizaines ou centaines d'instances gérées en un compartiment S3 central. Vous pouvez désormais chiffrer la synchronisation des données des ressources à l'aide d'une clé AWS Key Management Service. Pour plus d'informations, consultez la section Procédure : Utiliser la synchronisation de données de ressources pour regrouper les données d'inventaire (p. 843).	1 septembre 2017
Nouvelles procédures State Manager	Ajout de deux nouvelles procédures à la documentation State Manager : Mise à jour automatique d'Agent SSM (interface de ligne de commande) (p. 1043) Procédure : Mise à jour automatique des pilotes PV sur les instances EC2 pour Windows Server (console) (p. 1045)	31 août 2017

Modification	Description	Date de publication
Conformité de la configuration Systems Manager	Utilisez la conformité de configuration pour analyser votre parc d'instances gérées afin de rechercher des incohérences de conformité et de configuration de correctifs. Vous pouvez collecter et regrouper des données de plusieurs comptes et régions AWS, puis explorer plus en détails des ressources spécifiques qui ne sont pas conformes. Par défaut, le service Conformité de configuration affiche les données de conformité relatives Gestionnaire de correctifs à l'application des correctifs et State Manager aux associations. Vous pouvez également personnaliser le service et créer vos propres types de conformité en fonction de vos exigences métier ou informatiques. Pour plus d'informations, consultez la section Conformité de la configuration AWS Systems Manager (p. 778).	28 août 2017
Nouvelle action Automation : aws:executeAutomation	Exécute un flux de travail Automation secondaire en appelant un document Automation secondaire. Avec cette action, vous pouvez créer des documents Automation pour la plupart de vos flux de travail courants et faire référence à ces documents pendant une exécution d'Automation. Cette action peut simplifier vos documents Automation en supprimant la nécessité de dupliquer les étapes sur les documents similaires. Pour plus d'informations, consultez la section aws:executeAutomation – Exécuter une autre exécution d'automatisation (p. 414).	22 août 2017
Automation en tant que cible d'un événement CloudWatch	Vous pouvez démarrer un flux de travail Automation en spécifiant un document d'Automation en tant que cible d'un événement Amazon CloudWatch. Vous pouvez démarrer des workflows selon un programme ou lorsqu'un événement de système AWS spécifique se produit. Pour plus d'informations, consultez la section Exécution d'automatisations avec des déclencheurs à l'aide deEventBridge (p. 343).	21 août 2017
Gestion des versions et mises à jour générales des associations State Manager	Vous pouvez à présent créer différentes versions d'association State Manager. Il existe une limite de 1 000 versions pour chaque association. Vous pouvez également spécifier des noms pour vos associations. De même, la documentation State Manager a été mise à jour pour corriger les informations obsolètes et les incohérences. Pour plus d'informations, consultez la section AWS Systems Manager State Manager (p. 987).	21 août 2017

Modification	Description	Date de publication
Modifications apportées à Fenêtres de maintenance	Fenêtres de maintenance inclut les modifications et améliorations suivantes : • Auparavant, Fenêtres de maintenance ne pouvait effectuer de tâches qu'en utilisant Fonctionnalité Exécuter la commande. Vous pouvez désormais réaliser des tâches à l'aide de Systems Manager Automation, de AWS Lambda et de AWS Step Functions. • Vous pouvez modifier les cibles d'une fenêtre de maintenance, et spécifier le nom, la description et le propriétaire d'une cible. • Vous pouvez modifier les tâches d'une fenêtre de maintenance telles que la spécification d'un nouveau document SSM pour les tâches Fonctionnalité Exécuter la commande et Automation. • Tous les Fonctionnalité Exécuter la commande paramètres sont désormais pris en charge, y compris DocumentHash , DocumentHashType , , Comment et TimeoutSeconds .NotificationConfig • Vous pouvez désormais utiliser une balise `safe` lorsque vous tentez d'annuler l'inscription d'une cible. Si l'option est activée, le système renvoie une erreur lors du référencement d'une cible par toute tâche. Pour plus d'informations, consultez la section AWS Systems Manager Fenêtres de maintenance (p. 688).	16 août 2017
Nouvelle action Automation : aws:approve	Cette nouvelle action des documents Automation interrompt temporairement l'exécution d'Automation jusqu'à ce que les mandataires désignés aient approuvé ou rejeté l'action. Une fois le nombre d'approbations requises atteint, l'exécution d'Automation reprend. Pour plus d'informations, consultez la section Référence sur les actions Systems Manager Automation (p. 386).	10 août 2017

Modification	Description	Date de publication
Rôle responsable d'Automation plus requis	Auparavant, Automation nécessitait que vous spécifiiez un rôle de service (ou rôle responsable) afin que ce service ait la permission de réaliser des actions en votre nom. Automation ne nécessite plus ce rôle car le service opère désormais à l'aide du contexte de l'utilisateur ayant invoqué l'exécution. Néanmoins, les situations suivantes nécessitent tout de même que vous spécifiiez un rôle du service pour l'automatisation : • Lorsque vous souhaitez restreindre les privilèges d'un utilisateur sur une ressource tout en souhaitant que l'utilisateur puisse exécuter un flux de travail d'automatisation nécessitant des privilèges supérieurs. Dans ce scénario, vous pouvez créer un rôle de service avec des privilèges supérieurs et autoriser l'utilisateur à exécuter le flux de travail. • Les opérations qui seront probablement exécutées pendant plus de 12 heures nécessitent un rôle de service. Pour plus d'informations, consultez la section Configuration d'Automation (p. 308).	3 août 2017
Conformité de la configuration	Utilisez la conformité de configuration Amazon EC2 Systems Manager pour analyser votre parc d'instances gérées afin de rechercher des incohérences de conformité et de configuration de correctifs. Vous pouvez collecter et regrouper des données de plusieurs comptes et régions AWS, puis explorer plus en détails des ressources spécifiques qui ne sont pas conformes. Pour plus d'informations, consultez la section Conformité de la configuration AWS Systems Manager (p. 778).	8 août 2017
Améliorations de document SSM	Les documents de commande et de stratégie SSM offrent désormais le support multiplateforme. Cela signifie qu'un même document SSM, peut traiter les plugins pour les systèmes d'exploitation Windows et Linux. Le support multiplateforme vous permet de consolider le nombre de documents que vous gérez. Le support multiplateforme est proposé dans les documents SSM utilisant la version de schéma 2.2 ou ultérieure. SSM Les documents de commande utilisant la version de schéma 2.0 ou ultérieure peuvent désormais inclure plusieurs plugins du même type. Par exemple, vous pouvez créer un document de commande qui appelle le `aws:runRunShellScript` plug-in plusieurs fois. Pour de plus amples informations sur les modifications apportées à la version 2.2 du schéma, veuillez consulter Documents AWS Systems Manager (p. 1211). Pour de plus amples informations sur les plug-ins SSM, veuillez consulter Plug-ins Systems Manager.	12 juillet 2017

Modification	Description	Date de publication
Application des correctifs Linux	Gestionnaire de correctifspeut désormais corriger les distributions Linux suivantes : Systèmes 64 bits et 32 bits • Amazon Linux2014.03, 2014.09 ou version ultérieure • Ubuntu Server 16.04 LTS, 14.04 LTS ou 12.04 LTS • Red Hat Enterprise Linux(RHEL) 6.5 ou version ultérieure Systèmes 64 bits uniquement • Amazon Linux2015.03, 2015.09 ou version ultérieure • Red Hat Enterprise Linux(RHEL) 7.x ou version ultérieure Pour plus d'informations, consultez la section AWS Systems Manager Gestionnaire de correctifs (p. 1047). Note • Pour corriger les instances Linux, vos instances doivent exécuter Agent SSM version 2.0.834.0 ou ultérieure. Pour plus d'informations sur la mise à jour de l'agent, consultez la section intitulée Exemple : Mettre à jour Agent SSM dans Exécution des commandes depuis la console (p. 946) . • Le document `AWS-ApplyPatchBaseline` SSM est en cours de remplacement par le `AWS-RunPatchBaseline` document.	6 juillet 2017
Synchronisation de données de ressources	Vous pouvez utiliser la synchronisation de données de ressources Systems Manager pour envoyer les données d'inventaire collectées depuis toutes vos instances gérées vers un même compartiment Amazon S3. La synchronisation des données de ressource met alors automatiquement à jour les données centralisées lors de la collecte de nouvelles données d'inventaire. Avec toutes les données d'inventaire stockées dans un compartiment S3 cible, vous pouvez utiliser les services tels que Amazon Athena et Amazon QuickSight pour questionner et analyser les données regroupées. Pour de plus amples informations, veuillez consulter Configuration de la synchronisation de données de ressource pour l'inventaire (p. 800). Pour obtenir un exemple d'utilisation de la synchronisation des données de ressource, consultez Procédure : Utiliser la synchronisation de données de ressources pour regrouper les données d'inventaire (p. 843).	29 juin 2017

Modification	Description	Date de publication
Hiérarchies des paramètres Systems Manager	La gestion de douzaines ou de centaines de paramètres Systems Manager comme une liste simple est chronophage et propice aux erreurs. Vous pouvez utiliser les hiérarchies des paramètres pour vous aider à organiser et à gérer des paramètres Systems Manager. Une hiérarchie est un nom de paramètre qui comporte un chemin que vous définissez en utilisant des barres obliques. Voici un exemple qui utilise trois niveaux de hiérarchie dans le nom pour identifier ce qui suit : /Environnement/Type d'ordinateur/Application/Données `/Dev/DBServer/MySQL/db-string13` Pour plus d'informations, consultez la section Utilisation des hiérarchies de paramètres (p. 285). Pour obtenir un exemple d'utilisation des hiérarchies des paramètres, consultez Utilisation des hiérarchies de paramètres (p. 285).	22 juin 2017
Prise en charge de Agent SSM pour SUSE Linux Enterprise Server	Vous pouvez installer Agent SSM sur 64 bits SUSE Linux Enterprise Server (SLES). Pour plus d'informations, consultez la section Installation et configuration de l'Agent SSM sur les instances EC2 pour Linux (p. 75).	14 juin 2017

Glossaire AWS

Pour la terminologie AWS la plus récente, consultez le Glossaire AWS dans le document AWS General Reference.

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.