Guide de l’utilisateur

AWS Systems Manager

AWS Systems Manager: Guide de l’utilisateur

Table of Contents

Qu'est-ce que c'est AWS Systems Manager ?

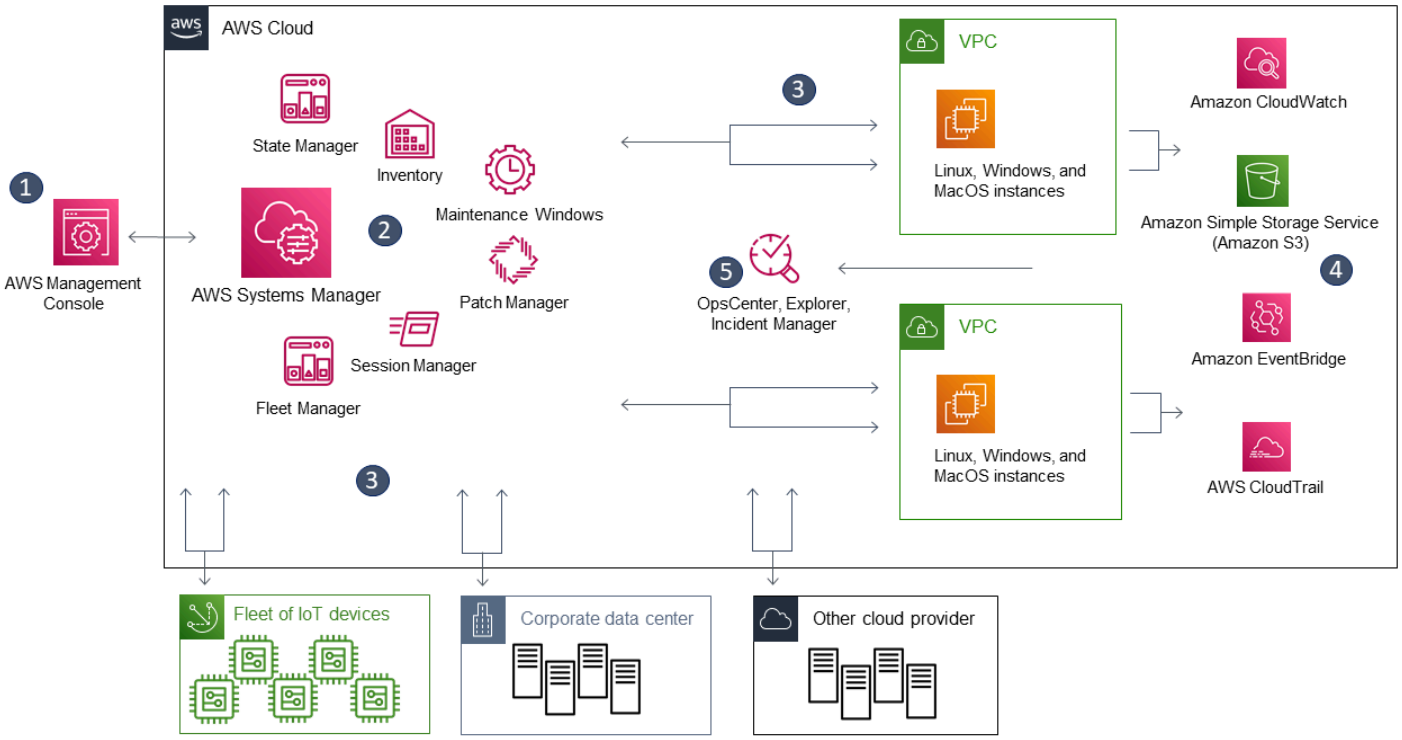
AWS Systems Manager est le centre des opérations pour vos AWS applications et vos ressources et une solution de end-to-end gestion sécurisée pour les environnements hybrides et multicloud qui permet des opérations sécurisées à grande échelle.

Fonctionnement de Systems Manager

Le schéma suivant décrit comment certaines capacités de Systems Manager effectuent des actions sur vos ressources. Le diagramme n'inclut pas toutes les fonctionnalités. Chaque interaction énumérée est décrite avant le diagramme.

1. Access Systems Manager : utilisez l'une des options disponibles pour l'accès à Systems Manager.
2. Choose a Systems Manager capability (Choisissez une fonctionnalité Systems Manager) : déterminez quelle fonctionnalité peut vous aider à effectuer l'action que vous souhaitez effectuer sur vos ressources. Le schéma ne montre que quelques-unes des fonctionnalités utilisées par les administrateurs informatiques et le DevOps personnel pour gérer leurs applications et leurs ressources.
3. Vérification et traitement : Systems Manager vérifie que votre utilisateur, groupe ou rôle dispose des autorisations AWS Identity and Access Management (IAM) requises pour effectuer l'action que vous avez spécifiée. Si la cible de votre action est un nœud géré, l'agent Systems Manager (SSM Agent) exécuté sur le nœud effectue l'action. Pour les autres types de ressources, Systems Manager exécute l'action spécifiée ou communique avec d'autres Services AWS personnes pour exécuter l'action au nom de Systems Manager.
4. Reporting (Génération de rapports) : état du rapport pour Systems Manager, SSM Agent et les autres Services AWS ayant effectué une action pour le compte de Systems Manager. Systems Manager peut envoyer des informations d'état à d'autres personnes Services AWS, s'il est configuré.
5. Systems Manager operations management capabilities (Fonctionnalités de gestion des opérations de Systems Manager) : si cette option est activée, les fonctionnalités de gestion des opérations de Systems Manager telles que Explorer OpsCenter, et Incident Manager agrège les données d'opérations ou crée des artefacts tels que des éléments de travail opérationnels (OpsItems) et des incidents en réponse à des événements ou des erreurs liés à vos ressources. Ces artefacts incluent des éléments de travail opérationnels (OpsItems) et des incidents. Les fonctionnalités de gestion des opérations de Systems Manager fournissent des informations opérationnelles sur vos

applications et ressources ainsi que des solutions de correction automatisées pour résoudre les problèmes.



Fonctions de Systems Manager

System Manager regroupe les fonctionnalités dans les catégories suivantes. Sélectionnez les onglets de chaque catégorie pour en savoir plus sur chaque fonctionnalité.

Rubriques

- Gestion des applications
- Gestion des modifications
- Gestion des nœuds
- Gestion des opérations
- Quick Setup
- Ressources partagées

Gestion des applications

Gestionnaire d'application

Application Manageraide DevOps les ingénieurs à étudier et à résoudre les problèmes liés à leurs AWS ressources dans le contexte de leurs applications et de leurs clusters. Dans Application Manager, une application est un groupe logique de ressources AWS que vous voulez exploiter en tant qu'unité. Ce groupe logique peut représenter différentes versions d'une application, les limites de propriété pour les opérateurs ou les environnements de développement, pour n'en citer que quelques-uns. Application Managerle support pour les clusters de conteneurs inclut à la fois les clusters Amazon Elastic Kubernetes Service (Amazon EKS) et Amazon Elastic Container Service (Amazon ECS). Application Managerregroupe les informations d'exploitation issues de plusieurs fonctionnalités Services AWS et de Systems Manager en une seule AWS Management Console.

AppConfig

AppConfig vous aide à créer, gérer et déployer des configurations d'applications et des drapeaux de fonctionnalités. AppConfig prend en charge les déploiements contrôlés vers des applications de toute taille. Vous pouvez utiliser AppConfig avec les applications hébergées sur des instances Amazon EC2, des conteneurs AWS Lambda , des applications mobiles ou des appareils de périphérie. Pour éviter les erreurs lors du déploiement de configurations d'application, AppConfig propose des validateurs. Un validateur permet une vérification syntaxique ou sémantique pour vérifier que la configuration que vous souhaitez déployer fonctionne comme prévu. Lors d'un déploiement de configuration, AppConfig surveille l'application pour vérifier 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 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 ID d'instance Amazon Elastic Compute Cloud (Amazon EC2), des ID Amazon Machine Image (AMI) et des codes de licence en tant que valeurs de paramètre. 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.

Gestion des modifications

Gestionnaire des modifications

Change Manager est un cadre de gestion des modifications d'entreprise pour demander, approuver, implémenter et signaler des modifications opérationnelles apportées à la configuration et à l'infrastructure de votre application. À partir d'un seul compte d'administrateur délégué, si vous en avez un AWS Organizations, vous pouvez gérer les modifications sur plusieurs Comptes AWS comptes Régions AWS. En variante, en utilisant un compte local, vous pouvez gérer les modifications d'un Compte AWS unique. Change ManagerÀ utiliser pour gérer les modifications apportées aux AWS ressources et aux ressources locales.

Automation

Utilisez Automation pour automatiser les tâches courantes de maintenance et de déploiement. Vous pouvez utiliser Automation pour créer et mettre à jour des Amazon Machine Images (AMIs), appliquer des mises à jour de pilote et d'agent, réinitialiser les mots de passe sur les instances Windows Server, réinitialiser les clés SSH sur les instances Linux, et appliquer des correctifs de OS ou des mises à jour d'application.

Change Calendar

Change Calendar vous permet de configurer des plages de dates et de temps dans lesquelles les actions que vous spécifiez (dans des runbooks Systems Manager Automation, par exemple) peuvent être effectuées ou non dans votre Compte AWS. Dans Change Calendar, ces plages sont appelées des événements. Lorsque vous créez une entrée Change Calendar, vous créez un document Systems Manager de type `ChangeCalendar`. Dans Change Calendar, le document stocke les données iCalendar 2.0 au format texte brut. Les événements que vous ajoutez à l'entrée Change Calendar font partie du document. Vous pouvez ajouter des événements manuellement dans l'interface Change Calendar ou importer des événements à partir d'un calendrier tiers pris en charge en utilisant un fichier `.ics`.

Fenêtres de maintenance

Utilisez les Maintenance Windows 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.

Gestion des nœuds

Un nœud géré est une machine configurée pour être utilisée avec Systems Manager dans des environnements hybrides et multicloud.

Compliance

Utilisez Compliance pour analyser votre flotte de nœuds gérés afin de rechercher des incohérences de conformité et de configuration de correctifs. Vous pouvez collecter et agréger des données provenant de plusieurs Comptes AWS sources Régions AWS, puis explorer des ressources spécifiques non conformes. Par défaut, le service Compliance affiche les données de conformité relatives aux correctifs Patch Manager et aux associations State Manager. Vous pouvez également personnaliser le service et créer vos propres types de conformité en fonction de vos exigences métier ou IT.

Fleet Manager

Fleet Manager est une expérience d'interface utilisateur (UI) unifiée qui vous aide à gérer à distance vos nœuds. Avec Fleet Manager, vous pouvez afficher l'état et le statut de performance de votre flotte à partir d'une console unique. Vous pouvez également collecter des données provenant d'appareils et d'instances individuels pour effectuer des tâches courantes de résolution des problèmes et de gestion à partir de la console. Cela comprend, entre autres, l'affichage du contenu des répertoires et des fichiers, la gestion du registre Windows, la gestion des utilisateurs du système d'exploitation.

Inventory

Inventory automatise le processus de collecte de l'inventaire logiciel à partir d'instances gérées. Vous pouvez utiliser Inventory pour collecter des métadonnées sur des applications, des fichiers, des composants, des correctifs et bien plus.

Gestionnaire de session

Utilisez-le Session Managerpour gérer vos appareils Edge et vos instances Amazon Elastic Compute Cloud (Amazon EC2) via un shell interactif basé sur un navigateur en un clic ou via le. AWS CLISession Managerfournit une gestion sécurisée et vérifiable des périphériques et des instances de périphérie sans avoir à ouvrir les ports entrants, à gérer les hôtes Bastion ou à gérer les clés SSH. Session Managervous permet également de vous conformer aux politiques de l'entreprise qui exigent un accès contrôlé aux appareils et aux instances Edge, des pratiques de sécurité strictes et des journaux entièrement vérifiables avec les détails d'accès aux appareils et aux instances Edge, tout en fournissant aux utilisateurs finaux un accès multiplateforme en

un clic à vos appareils Edge et à vos instances EC2. Pour utiliser Session Manager, vous devez activer le niveau d'instances avancées. Pour plus d'informations, consultez Activation du niveau d'instances avancées.

Fonctionnalité Exécuter la commande

Utilisez Run Command pour gérer à distance et en toute sécurité la configuration de vos nœuds gérés à grande échelle. Utilisez Run Command pour effectuer des modifications à la demande, comme la mise à jour d'applications ou l'exécution de scripts shell Linux et de commandes Windows PowerShell sur un ensemble cible de dizaines ou de centaines de nœuds gérés.

State Manager

Utilisez State Manager pour automatiser le processus permettant de conserver vos nœuds gérés à un état défini. Vous pouvez utiliser State Manager pour vous assurer que vos nœuds gérés sont amorcés avec un logiciel spécifique au démarrage, joints à un domaine Windows (nœuds Windows Server uniquement) ou corrigés avec des mises à jour logicielles spécifiques.

Gestionnaire de correctifs

Utilisez Patch Manager pour automatiser le processus d'application des correctifs aux nœuds gérés, avec les mises à jour liées à la sécurité et autres. Vous pouvez utiliser Patch Manager 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 publiées par Microsoft.)

Cette fonctionnalité vous permet de rechercher des correctifs manquants dans les nœuds gérés et d'appliquer les correctifs manquants individuellement ou à de grands groupes de nœuds gérés à l'aide de balises. Patch Manager utilise des référentiels de correctifs, qui peuvent inclure des règles d'approbation automatique des correctifs quelques jours après leur publication, ainsi qu'une liste des correctifs approuvés et refusés. Vous pouvez installer des correctifs de sécurité régulièrement en planifiant leur exécution sous forme de tâches de fenêtre de maintenance Systems Manager, ou bien vous pouvez corriger vos nœuds gérés 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 référentiel 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 le nœud géré. Pour Linux, vous avez également la possibilité de mettre à jour n'importe quel package sur le nœud géré, pas seulement ceux qui sont classés en tant que mises à jour de sécurité du système d'exploitation. Vous pouvez également générer des rapports des correctifs, qui sont envoyés à un compartiment

Amazon S3 de votre choix. Pour un nœud géré individuel, les rapports contiennent les détails de tous les correctifs relatifs à la machine. Pour un ensemble de nœuds gérés, le rapport contient seulement un résumé indiquant le nombre de correctifs manquants.

Distributor

Utilisez Distributor pour créer et déployer des packages sur des nœuds gérés. Vous pouvez ainsi créer votre propre package logiciel (ou rechercher des packages logiciels d'agent AWS fournis, tels que AmazonCloudWatchAgent) à installer sur les nœuds gérés par Systems Manager. Distributor Après avoir installé un package pour la première fois, vous pouvez utiliser Distributor pour désinstaller et réinstaller une nouvelle version du package, ou effectuer une mise à jour sur place qui ajoute des fichiers nouveaux ou modifiés. Distributor publie des ressources, telles que des packages logiciels, sur des nœuds gérés par Systems Manager.

Hybrid Activations

Pour configurer des machines non EC2 dans votre environnement hybride et multicloud en tant que nœuds gérés, créez une activation hybride. 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 Amazon Elastic Compute Cloud (Amazon EC2) et une clé secrète pour fournir un accès sécurisé au service Systems Manager à partir de vos instances gérées.

Vous pouvez également créer une activation pour les appareils de périphérie si vous souhaitez les gérer à l'aide de Systems Manager.

Gestion des opérations

Incident Manager

Incident Manager est une console de gestion des incidents qui aide les utilisateurs à atténuer les incidents affectant leurs applications AWS hébergées et à s'en remettre.

Incident Manager améliore la résolution des incidents en informant les intervenants de l'impact, en mettant en évidence les données de dépannage pertinentes et en fournissant des outils de collaboration pour assurer la sauvegarde et l'exécution des services. Incident Manager automatise également les plans de réponse et fait remonter les problèmes à l'équipe concernée.

Explorer

Explorerest un tableau de bord des opérations personnalisable qui fournit des informations sur vos AWS ressources. Exploreraffiche une vue agrégée des données d'exploitation (OpsData)

pour vous Comptes AWS et pour l'ensemble de celles-ci Régions AWS. DansExplorer, OpsData inclut les métadonnées relatives à vos instances Amazon EC2, les détails de conformité des correctifs et les éléments de travail opérationnels ()OpsItems. Explorerfournit un contexte sur la manière dont elles OpsItems sont réparties entre vos unités commerciales ou vos applications, sur leur évolution dans le temps et sur 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 OpsCenter, une des fonctionnalités de Systems Manager, pour exécuter des runbooks Automation et résoudre ces problèmes.

OpsCenter

OpsCenterfournit un emplacement central où les ingénieurs des opérations et IT les professionnels peuvent consulter, étudier et résoudre les éléments de travail opérationnels (OpsItems) liés aux AWS ressources. OpsCenterest conçu pour réduire le délai moyen de résolution des problèmes ayant une incidence sur les AWS ressources. 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 runbooks Systems Manager Automation que vous pouvez utiliser pour résoudre les problèmes. Vous pouvez spécifier des données personnalisées consultables pour chaque OpsItem. Vous pouvez également afficher des rapports de synthèse générés automatiquement sur les OpsItems par statut et source.

CloudWatch Dashboards

Les CloudWatch tableaux de bord Amazon sont des pages personnalisables de la CloudWatch console que vous pouvez utiliser pour surveiller vos ressources dans une vue unique, même celles qui sont réparties dans différentes régions. Vous pouvez utiliser CloudWatch des tableaux de bord pour créer des vues personnalisées des mesures et des alarmes relatives à vos AWS ressources.

Quick Setup

Quick SetupÀ utiliser pour configurer Services AWS les fonctionnalités fréquemment utilisées conformément aux meilleures pratiques recommandées. Vous pouvez l'utiliser de Quick Setup manière individuelle Compte AWS ou multiple Comptes AWS et Régions AWS en intégrant à AWS Organizations. Quick Setupsimplifie la configuration des services, y compris Systems Manager, en automatisant les tâches courantes ou recommandées. Ces tâches incluent, par exemple, la création de rôles de profil d'instance AWS Identity and Access Management (IAM) requis et la mise en place

de meilleures pratiques opérationnelles, telles que des analyses de correctifs périodiques et la collecte d'inventaire.

Ressources partagées

Documents

Un document Systems Manager (document SSM) définit les actions exécutées par Systems Manager. Les types de documents SSM incluent des documents Command, qui sont utilisés par State Manager et Run Command, et des runbooks Automation, qui sont utilisés par Systems Manager Automation. Systems Manager inclut 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.

Accès à Systems Manager

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

Console Systems Manager

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

AWS IoT Greengrass V2 console

Vous pouvez afficher et gérer les appareils périphériques configurés pour AWS IoT Greengrass la console Greengrass.

AWS outils de ligne de commande

À l'aide des outils de ligne de AWS commande, vous pouvez émettre des commandes sur la ligne de commande de votre système pour exécuter Systems Manager et d'autres AWS tâches. Les outils sont pris en charge sous Linux, macOS et Windows. Utiliser la AWS Command Line Interface (AWS CLI) peut être plus rapide et plus pratique que d'utiliser la console. Les outils de ligne de commande sont également utiles si vous souhaitez créer des scripts exécutant des tâches AWS .

AWS fournit deux ensembles d'outils de ligne de commande : le AWS Command Line Interfaceet le AWS Tools for Windows PowerShell. Pour plus d'informations sur l'installation et l'utilisation du AWS CLI, consultez le guide de AWS Command Line Interface l'utilisateur. Pour plus

d'informations sur l'installation et l'utilisation des outils pour Windows PowerShell, consultez le guide de AWS Tools for Windows PowerShell l'utilisateur.

> **Note**
>
> Sur vos instances Windows Server, Windows PowerShell 3.0 ou une version ultérieure est requis pour exécuter certains documents SSM (par exemple, le document `AWS-ApplyPatchBaseline` hérité). Vérifiez que votre instance Windows Server exécute actuellement Windows Management Framework 3.0 ou une version ultérieure. Le cadre inclut Windows PowerShell.

AWS SDK

AWS fournit des kits de développement logiciel (SDK) composés de bibliothèques et d'exemples de code pour divers langages de programmation et plateformes (par exemple, Java, Python, Ruby, .NET, iOS et Android, etc.). Les SDK facilitent l'octroi par programmation d'un accès à Systems Manager. Pour plus d'informations sur les AWS SDK, notamment sur la façon de les télécharger et de les installer, consultez la section Outils pour Amazon Web Services.

Historique des noms de service Systems Manager

AWS Systems Manager (Systems Manager) était auparavant connu sous les noms de Amazon Simple Systems Manager (SSM) « » et « Amazon EC2 Systems Manager (SSM) ». Le nom abrégé original du service, SSM « », est toujours reflété dans diverses AWS ressources, y compris quelques autres consoles de service. Voici quelques exemples :

- Systems Manager Agent : SSM Agent
- Paramètres Systems Manager : paramètres SSM
- Points de terminaison de service Systems Manager : `ssm.region.amazonaws.com`
- AWS CloudFormation types de ressources : `AWS::SSM::Document`
- AWS Config identifiant de règle : `EC2_INSTANCE_MANAGED_BY_SSM`
- AWS Command Line Interface (AWS CLI) commandes : `aws ssm describe-patch-baselines`
- AWS Identity and Access Management Noms des politiques gérées (IAM) : `AmazonSSMReadOnlyAccess`

- ARN de la ressource Systems Manager : `arn:aws:ssm:region:account-id:patchbaseline/pb-07d8884178EXAMPLE`

Soutenu Régions AWS

Systems Manager est disponible dans la Régions AWS liste des points de terminaison de service Systems Manager du Référence générale d'Amazon Web Services. Avant de démarrer le processus de configuration de Systems Manager, nous vous recommandons de vérifier que le service est disponible dans chacun des Régions AWS sites dans lesquels vous souhaitez l'utiliser.

Pour les machines non EC2 de votre environnement hybride et multicloud, nous vous conseillons de choisir la région plus proche de votre centre de données ou de votre environnement informatique.

Systèmes d'exploitation et types de machines pris en charge

Avant d'utiliser Systems Manager, vérifiez que votre système d'exploitation, sa version et le type de machine sont pris en charge en tant que nœuds gérés.

Rubriques

- Systèmes d'exploitation pris en charge pour Systems Manager
- Types de machines pris en charge dans les environnements hybrides et multicloud

Systèmes d'exploitation pris en charge pour Systems Manager

Les sections suivantes répertorient les systèmes d'exploitation et leurs versions pris en charge par Systems Manager.

> **Note**
>
> Si vous envisagez de gérer et de configurer des appareils AWS IoT Greengrass principaux à l'aide de Systems Manager, ces appareils doivent répondre aux exigences de AWS IoT Greengrass. Pour plus d'informations, consultez la section Configuration des appareils AWS IoT Greengrass principaux dans le Guide du AWS IoT Greengrass Version 2 développeur. Si vous envisagez de gérer et de configurer AWS IoT des appareils non AWS périphériques, ces appareils doivent répondre aux exigences répertoriées ici et être configurés en tant que

nœuds gérés sur site pour Systems Manager. Pour plus d'informations, consultez Gestion des appareils de pointe avec Systems Manager.

Important

Patch Manager, une fonctionnalité de Systems Manager, peut ne pas prendre en charge toutes les versions du système d'exploitation répertoriées dans cette rubrique. Afin d'obtenir la liste des versions du système d'exploitation prises en charges par Patch Manager, consultez Conditions préalables requises Patch Manager.

Types de système d'exploitation

- Linux
- macOS (Instances Amazon EC2 uniquement)
- Raspberry Pi OS (anciennement Raspbian)
- Windows Server

Linux

AlmaLinux

Versions	x86	x86_64	ARM64
8,3—8,9		✓	✓
9.0 à 9.2		✓	✓

Amazon Linux 1

Versions	x86	x86_64	ARM64
2012.03 à 2018.03	✓	✓	

> **Note**
>
> À partir de la version 2015.03, Amazon Linux 1 a été publié en x86_64 plusieurs versions. Amazon Linux 1 a atteint la fin de son support standard le 31 décembre 2020 et sa fin de vie le 31 décembre 2023, comme annoncé dans la mise à jour sur Amazon Linux AMI end-of-life sur le blog d'AWS actualités. AWS ne fournit plus Amazon Machine Images (AMIs) pour ce système d'exploitation. AWS Systems Manager continue toutefois de fournir un support pour les instances Amazon Linux 1 existantes.

Amazon Linux 2

Versions	x86	x86_64	ARM64
2.0 et toutes les versions ultérieures		✓	✓

Amazon Linux 2023

Versions	x86	x86_64	ARM64
2023.0.20230315.0 et toutes les versions ultérieures		✓	✓

Bottlerocket

Versions	x86_64	ARM64
1.0.0 et toutes les versions ultérieures	✓	✓

CentOS

Versions	x86	x86_64	ARM64
6.x[1]	✓	✓	

Versions	x86	x86_64	ARM64
7.1 et versions 7.x ultérieures		✓	✓
8.0 à 8.5		✓	✓

¹ Pour utiliser ces versions, vous devez utiliser une version 3.0.x de SSM Agent. Nous recommandons d'utiliser la dernière version disponible 3.0.x de SSM Agent. Les versions ultérieures de SSM Agent (3.1 ou version ultérieure) ne sont pas prises en charge.

CentOS Stream

Versions	x86	x86_64	ARM64
8		✓	✓

Debian Server

Versions	x86	x86_64	ARM64
Jessie (8)		✓	
Stretch (9)		✓	✓
Buster (10)		✓	✓
Bullseye (11)		✓	✓
Bookworm(12)		✓	✓

Oracle Linux

Versions	x86	x86_64	ARM64
7.5 à 7.8		✓	
8,1 à 8,9		✓	

Versions	x86	x86_64	ARM64
9.0 à 9.2		✓	

Red Hat Enterprise Linux (RHEL)

Versions	x86	x86_64	ARM64
6.x[1]	✓	✓	
7.0 à 7.5		✓	
7,6—8,9		✓	✓
9,0—9,3		✓	✓

[1] Pour utiliser ces versions, vous devez utiliser une version 3.0.x de SSM Agent. Nous recommandons d'utiliser la dernière version disponible 3.0.x de SSM Agent. Les versions ultérieures de SSM Agent (3.1 ou version ultérieure) ne sont pas prises en charge.

Rocky Linux

Versions	x86	x86_64	ARM64
8,4—8,9		✓	✓
9.0 à 9.2		✓	✓

SUSE Linux Enterprise Server (SLES)

Versions	x86	x86_64	ARM64
12 et versions 12.x ultérieures		✓	
15 et versions 15.x ultérieures		✓	✓

Ubuntu Server

Versions	x86	x86_64	ARM64
12.04 LTS et 14.04 LTS	✓	✓	
16.04 LTS et 18.04 LTS		✓	✓
20.04 LTS et 20.10 STR		✓	✓
22.04 LTS		✓	✓
23,04		✓	✓

macOS (Instances Amazon EC2 uniquement)

Version	x86	x86_64	Mac with Apple silicon
10.14.x (Mojave)		✓	
10.15.x (Catalina)		✓	
11.x (Big Sur)		✓	✓
12.x (Monterey)		✓	✓
13.x (Ventura)		✓	✓
14.x (Sonoma)		✓	✓

> **Note**
>
> macOSn'est pas pris en charge dans tous les cas Régions AWS. Pour plus d'informations sur la prise en charge d'Amazon EC2 pourmacOS, consultez les instances Mac Amazon EC2 dans le guide de l'utilisateur Amazon EC2.

Raspberry Pi OS (anciennement Raspbian)

Version	ARM32
8 (Jessie)	✓
9 (Stretch)	✓

Plus d'informations

- Gérer les périphériques Raspberry Pi avec AWS Systems Manager

Windows Server

SSM Agentnécessite Windows PowerShell 3.0 ou ultérieurement d'exécuter certains AWS Systems Manager documents (documents SSM) sur des Windows Server instances (par exemple, le `AWS-ApplyPatchBaseline` document existant). Vérifiez que votre instance Windows Server exécute actuellement Windows Management Framework 3.0 ou une version ultérieure. Ce cadre inclut Windows PowerShell. Pour de plus amples informations, consultez Windows Management Framework 3.0.

Version	x86	x86_64	ARM64
2008[1]	✓	✓	
2008 R2[1]		✓	
2012 et 2012 R2		✓	
2016		✓	
2019		✓	
2022		✓	

[1] À compter du 14 janvier 2020, Windows Server 2008 n'est plus pris en charge pour les mises à jour de fonctions ou de sécurité de Microsoft. Les Amazon Machine Images (AMIs) héritées, pour Windows Server 2008 et 2008 R2, incluent toujours la version 2 de l'SSM Agent, mais Systems

Manager ne prend plus officiellement en charge les versions 2008 et ne met plus à jour l'agent pour ces versions de Windows Server. En outre, SSM Agent version 3 peut ne pas être compatible avec toutes les opérations sur Windows Server 2008 et 2008 R2. La version finale officiellement prise en charge de l'SSM Agent pour les versions Windows Server 2008 est 2.3.1644.0.

Types de machines pris en charge dans les environnements hybrides et multicloud

Systems Manager prend en charge un certain nombre de types de machines en tant que nœuds gérés. Un nœud géré est une machine configurée pour fonctionner avec Systems Manager.

Ce guide de l'utilisateur utilise les termes hybride et multicloud pour désigner un environnement contenant n'importe quelle combinaison des types de machines suivants :

- Instances Amazon Elastic Compute Cloud (Amazon EC2)
- Serveurs sur votre propre site (serveurs sur site)
- AWS IoT Greengrass appareils principaux
- AWS IoT et appareils non AWS périphériques
- Machines virtuelles (VM), y compris les VM dans d'autres environnements cloud

Pour plus d'informations sur la AWS prise en charge des environnements hybrides et multicloud, consultez la section AWS Solutions pour les environnements hybrides et multicloud.

Utilisation de Systems Manager avec un AWS SDK

AWS des kits de développement logiciel (SDK) sont disponibles pour de nombreux langages de programmation populaires. Chaque SDK fournit une API, des exemples de code et de la documentation qui facilitent la création d'applications par les développeurs dans leur langage préféré.

Documentation SDK	Exemples de code
AWS SDK for C++	AWS SDK for C++ exemples de code
AWS CLI	AWS CLI exemples de code
AWS SDK for Go	AWS SDK for Go exemples de code

Documentation SDK	Exemples de code
AWS SDK for Java	AWS SDK for Java exemples de code
AWS SDK for JavaScript	AWS SDK for JavaScript exemples de code
Kit AWS SDK pour Kotlin	Kit AWS SDK pour Kotlin exemples de code
AWS SDK for .NET	AWS SDK for .NET exemples de code
AWS SDK for PHP	AWS SDK for PHP exemples de code
AWS Tools for PowerShell	Outils pour des exemples PowerShell de code
AWS SDK for Python (Boto3)	AWS SDK for Python (Boto3) exemples de code
AWS SDK for Ruby	AWS SDK for Ruby exemples de code
Kit AWS SDK pour Rust	Kit AWS SDK pour Rust exemples de code
AWS SDK pour SAP ABAP	AWS SDK pour SAP ABAP exemples de code
Kit AWS SDK pour Swift	Kit AWS SDK pour Swift exemples de code

Exemple de disponibilité

Vous n'avez pas trouvé ce dont vous avez besoin ? Demandez un exemple de code en utilisant le lien Provide feedback (Fournir un commentaire) en bas de cette page.

Con AWS Systems Manager figuration

Effectuez les tâches décrites dans cette section pour configurer des rôles, des comptes d'utilisateurs, des autorisations et des ressources initiales pour AWS Systems Manager. Les tâches décrites dans cette section sont généralement effectuées par Compte AWS des administrateurs système. Une fois ces étapes terminées, les utilisateurs de votre organisation peuvent utiliser Systems Manager pour configurer, gérer et accéder à vos nœuds gérés. Un nœud géré est une machine configurée pour être utilisée avec Systems Manager dans un environnement hybride et multicloud.

> **Note**
>
> Si vous prévoyez d'utiliser des instances Amazon EC2 et vos propres ressources de calcul dans un environnement hybride et multicloud, commencez par suivre les étapes décrites dans Utilisation de Systems Manager avec des instances EC2. Cette rubrique présente ces étapes dans le meilleur ordre pour terminer la configuration de Systems Manager pour les instances EC2 et les machines non EC2.

Si vous en utilisez déjà un autre 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.

Rubriques

- Utilisation de Systems Manager avec des instances EC2
- Utilisation de Systems Manager dans des environnements hybrides et multicloud
- Gestion des appareils de pointe avec Systems Manager
- Création d'un administrateur AWS Organizations délégué pour Systems Manager
- Configuration générale pour AWS Systems Manager

Utilisation de Systems Manager avec des instances EC2

Effectuez les tâches décrites dans cette section pour configurer et configurer les rôles, les autorisations et les ressources initiales pour AWS Systems Manager. Les tâches décrites dans cette section sont généralement exécutées par les administrateurs de Compte AWS et système. Une fois

ces étapes terminées, les utilisateurs de votre organisation peuvent utiliser Systems Manager pour configurer, gérer et accéder aux instances Amazon Elastic Compute Cloud (Amazon EC2).

> **Note**
>
> Si vous prévoyez d'utiliser Systems Manager pour gérer et configurer des machines sur site, suivez les étapes décrites dans Utilisation de Systems Manager dans des environnements hybrides et multicloud. Si vous prévoyez d'utiliser des instances Amazon EC2 et des machines non EC2 dans un environnement hybride et multicloud, commencez par suivre les étapes décrites ici. Cette section présente les étapes dans l'ordre recommandé pour configurer les rôles, les utilisateurs, les autorisations et les ressources initiales à utiliser dans vos opérations Systems Manager.

Si vous en utilisez déjà un autre 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.

Table des matières

Configurer les autorisations d'instance requises pour Systems Manager

Par défaut, AWS Systems Manager n'est pas autorisé à effectuer des actions sur vos instances. Vous pouvez fournir des autorisations d'instance au niveau du compte à l'aide d'un rôle AWS Identity and Access Management (IAM) ou au niveau de l'instance à l'aide d'un profil d'instance. Si votre cas d'utilisation le permet, nous vous recommandons d'accorder l'accès au niveau du compte à l'aide de la configuration de gestion des hôtes par défaut.

Configuration recommandée pour les autorisations d'instance EC2

La configuration de gestion des hôtes par défaut permet à Systems Manager de gérer vos instances Amazon EC2 automatiquement. Une fois ce paramètre activé, toutes les instances utilisant le service de métadonnées d'instance version 2 (IMDSv2) dans Région AWS et Compte AWS avec

la SSM Agent version 3.2.582.0 ou ultérieure installée deviennent automatiquement des instances gérées. La configuration de gestion des hôtes par défaut ne prend pas en charge du Service des métadonnées d'instance Version 1. Pour plus d'informations sur la transition vers IMDSv2, consultez la section Transition vers l'utilisation du service de métadonnées d'instance version 2 dans le guide de l'utilisateur Amazon EC2. Pour plus d'informations sur la vérification de la version de l'SSM Agent installée sur votre instance, consultez Vérification du numéro de version de l'SSM Agent. Pour plus d'informations sur la mise à jour de l'SSM Agent, consultez Mise à jour automatique de l'SSM Agent. Les avantages des instances gérées sont les suivants :

- Connectez-vous à vos instances en toute sécurité à l'aide de Session Manager.
- Effectuez des analyses de correctifs automatisées à l'aide de Patch Manager.
- Consultez les informations détaillées sur vos instances à l'aide de Systems Manager Inventory.
- Suivez et gérez les instances à l'aide de Fleet Manager.
- Maintenez le SSM Agent à jour automatiquement.

Fleet Manager, InventairePatch Manager, et Session Manager sont des fonctionnalités de AWS Systems Manager.

La configuration de gestion des hôtes par défaut permet de gérer les instances sans utiliser de profils d'instance et garantit que Systems Manager dispose des autorisations nécessaires pour gérer toutes les instances de la région et du compte. Si les autorisations fournies ne sont pas suffisantes pour votre cas d'utilisation, vous pouvez également ajouter des politiques au rôle IAM par défaut créé par la configuration de gestion des hôtes par défaut. Sinon, si vous n'avez pas besoin d'autorisations pour toutes les fonctionnalités fournies par le rôle IAM par défaut, vous pouvez créer vos propres rôles et politiques personnalisés. Toutes les modifications apportées au rôle IAM que vous choisissez pour la configuration de gestion des hôtes par défaut s'appliquent à toutes les instances Amazon EC2 gérées dans la région et le compte. Pour plus d'informations sur la politique utilisée par la configuration de gestion des hôtes par défaut, consultez AWS stratégie gérée : politique InstanceDefault AmazonSSMManageDec2. Pour plus d'informations sur la configuration de gestion des hôtes par défaut, consultez Utilisation du paramètre de configuration de gestion d'hôte par défaut.

> **Important**
>
> Les instances enregistrées à l'aide de la configuration de gestion des hôtes par défaut stockent des informations d'enregistrement localement dans les répertoires `/lib/amazon/ssm` ou `C:\ProgramData\Amazon`. La suppression de ces répertoires ou de leurs fichiers

empêchera l'instance d'acquérir les informations d'identification nécessaires pour se connecter à Systems Manager à l'aide de la configuration de gestion des hôtes par défaut. Dans ces cas, vous devez utiliser un profil d'instance, pour fournir les autorisations requises à votre instance, ou recréer l'instance.

Note

Cette procédure est destinée à être exécutée uniquement par les administrateurs. Implémentez l'accès au moindre privilège lorsque vous autorisez des individus à configurer ou à modifier la configuration de gestion des hôtes par défaut. Vous devez activer la configuration de gestion d'hôte par défaut dans chaque instance pour laquelle Région AWS vous souhaitez gérer automatiquement vos instances Amazon EC2.

Activation du paramètre de configuration de gestion des hôtes par défaut

Vous pouvez activer la configuration de gestion des hôtes par défaut depuis la console Fleet Manager. Pour mener à bien cette procédure à l'aide de l'outil de ligne de commande AWS Management Console ou de votre outil de ligne de commande préféré, vous devez disposer des autorisations nécessaires pour les GetService opérations d'API ResetServiceSetting, UpdateServiceSetting et Setting. En outre, vous devez disposer des autorisations nécessaires pour l'autorisation `iam:PassRole` du rôle IAM `AWSSystemsManagerDefaultEC2InstanceManagementRole`. Voici un exemple de politique . Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations.

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Effect": "Allow",
   "Action": [
    "ssm:GetServiceSetting",
    "ssm:ResetServiceSetting",
    "ssm:UpdateServiceSetting"
   ],
   "Resource": "arn:aws:ssm:region:account-id:servicesetting/ssm/managed-instance/
default-ec2-instance-management-role"
```

```
  },
  {
   "Effect": "Allow",
   "Action": [
    "iam:PassRole"
   ],
   "Resource": "arn:aws:iam::account-id:role/service-role/
AWSSystemsManagerDefaultEC2InstanceManagementRole",
   "Condition": {
    "StringEquals": {
     "iam:PassedToService": [
      "ssm.amazonaws.com"
     ]
    }
   }
  }
 ]
}
```

Avant de commencer, si des profils d'instance sont associés à vos instances Amazon EC2, supprimez toutes les autorisations qui autorisent l'opération `ssm:UpdateInstanceInformation`. L'SSM Agent essaie d'utiliser les autorisations de profil d'instance avant d'utiliser les autorisations de configuration de gestion des hôtes par défaut. Si vous autorisez l'opération `ssm:UpdateInstanceInformation` dans vos profils d'instance, l'instance n'utilisera pas les autorisations de configuration de gestion des hôtes par défaut.

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Choisissez Paramétrer la configuration de gestion des hôtes par défaut dans la liste déroulante Gestion des comptes.
4. Activez l'option Activer la configuration de gestion des hôtes par défaut.
5. Choisissez le rôle IAM utilisé pour activer les fonctionnalités de Systems Manager pour vos instances. Nous vous recommandons d'utiliser le rôle par défaut dans Configuration de gestion des hôtes par défaut. Il contient l'ensemble des autorisations minimum pour gérer vos instances Amazon EC2 à l'aide de Systems Manager. Si vous préférez utiliser un rôle personnalisé, la politique de confiance du rôle doit autoriser Systems Manager en tant qu'entité de confiance.
6. Choisissez Configurer pour terminer la configuration.

Après avoir activé la Configuration de gestion des hôtes par défaut, 30 minutes peuvent s'écouler avant que vos instances n'utilisent les informations d'identification du rôle que vous avez choisi. Vous devez activer la Configuration de gestion des hôtes par défaut dans chaque région dans laquelle vous souhaitez gérer automatiquement vos instances Amazon EC2.

Configuration alternative pour les autorisations d'instance EC2

Vous pouvez accorder l'accès au niveau des instances individuelles à l'aide d'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 politiques IAM qui définissent les autorisations nécessaires pour un nouveau rôle ou un rôle que vous avez déjà créé.

> **Note**
>
> Vous pouvez utiliser Quick Setup une fonctionnalité de AWS Systems Manager pour configurer rapidement un profil d'instance sur toutes les instances de votre Compte AWS. Quick Setupcrée également un rôle de service IAM (ou assume un rôle), qui permet à Systems Manager d'exécuter des commandes en toute sécurité sur vos instances en votre nom. Quick Setup vous permet d'ignorer cette étape (étape 3), ainsi que l'étape 4. Pour plus d'informations, consultez AWS Systems Manager Quick Setup.

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

- Si vous configurez des machines non EC2 dans un environnement hybride et multicloud pour Systems Manager, vous n'avez pas besoin de leur créer de profil d'instance. À la place, configurez vos machines virtuelles et serveurs pour qu'ils puissent utiliser un rôle de service IAM. Pour plus d'informations, voir Créer le rôle de service IAM requis pour Systems Manager dans les environnements hybrides et multicloud.
- Si vous modifiez le profil d'instance IAM, l'actualisation des informations d'identification de l'instance peut prendre un peu de temps. L'SSM Agent 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'SSM Agent ou redémarrer l'instance.

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, sélectionnez Rôles, puis Créer un rôle.
3. Pour Trusted entity (Entité de confiance), choisissez Service AWS.
4. Directement sous Use case (Cas d'utilisation), choisissez EC2, puis Next (Suivant).
5. Sur la page Add permissions (Ajouter des autorisations), procédez comme suit :

   - Utilisez le champ de recherche pour trouver la politique ManagedInstanceprincipale d'AmazonSSM. Cochez la case en regard de son nom.

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

   - Si vous avez créé une politique de compartiment S3 personnalisée au cours de la procédure précédente, (Facultatif) créer une politique personnalisée pour l'accès au compartiment S3, recherchez-la et cochez la case en regard de son nom.
   - Si vous envisagez de joindre des instances à un Active Directory géré par AWS Directory Service, recherchez AmazonSSM DirectoryService Access et cochez la case à côté de son nom.
   - Si vous prévoyez d'utiliser EventBridge ou CloudWatch Logs pour gérer ou surveiller votre instance, recherchez CloudWatchAgentServerPolicy et cochez la case à côté de son nom.
6. Choisissez Suivant.
7. Pour Role name (Nom du rôle), saisissez un nom pour votre nouveau profil d'instance, par exemple **`SSMInstanceProfile`**.

   > **Note**
   >
   > Notez le nom de rôle. Vous pouvez choisir ce rôle lorsque vous créerez de nouvelles instances à gérer à l'aide de Systems Manager.

8. (Facultatif) Pour Description, mettez à jour la description pour ce profil d'instance.

9. (Facultatif) Pour Tags (Balises), ajoutez une ou plusieurs paires clé-valeur d'identification afin d'organiser, de suivre ou de contrôler l'accès pour ce rôle, puis sélectionnez Create role (Créer le 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 volet de navigation, sélectionnez Roles (Rôles), puis choisissez le rôle existant à associer à un profil d'instance pour des opérations Systems Manager.
3. Sous l'onglet Permissions (Autorisations), choisissez Add Permissions, Attach policies (Ajouter des autorisations, Attacher des politiques).
4. Sur la page Attach policy (Attacher la politique), procédez comme suit :
   - Utilisez le champ de recherche pour trouver la politique ManagedInstanceprincipale d'AmazonSSM. Cochez la case en regard de son nom.
   - Si vous avez créé une politique de compartiment S3 personnalisée, recherchez-la et cochez la case en regard de son nom. Pour plus d'informations sur les politiques de compartiment S3 personnalisés pour un profil d'instance, consultez (Facultatif) créer une politique personnalisée pour l'accès au compartiment S3.
   - Si vous envisagez de joindre des instances à un Active Directory géré par AWS Directory Service, recherchez AmazonSSM DirectoryService Access et cochez la case à côté de son nom.
   - Si vous prévoyez d'utiliser EventBridge ou CloudWatch Logs pour gérer ou surveiller votre instance, recherchez CloudWatchAgentServerPolicy et cochez la case à côté de son nom.
5. Choisissez Attach Policies (Attacher des politiques).

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 Guide de l'utilisateur IAM.

(Facultatif) créer une politique personnalisée pour l'accès au compartiment S3

La création d'une politique 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. Vous pouvez associer cette politique au rôle IAM par défaut créé par la Configuration de gestion des hôtes par défaut ou à un profil d'instance que vous avez créé lors de la procédure précédente.

Pour plus d'informations sur les compartiments S3 AWS gérés auxquels vous donnez accès dans le cadre de la politique suivante, consultezCommunications de l'SSM Agent avec des compartiments S3 gérés par AWS.

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le volet de navigation, sélectionnez Politiques, puis Créer une politique.
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 Resource (Ressource). Votre utilisateur, groupe ou rôle 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 élément `Statement`, remplacez chaque espace réservé *`region`* par l'identifiant de la Région AWS dans laquelle cette politique sera utilisée. Par exemple, utilisez `us-east-2` pour la région USA Est (Ohio). Pour obtenir une liste des valeurs *`region`* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la Référence générale d'Amazon Web Services.

   > **Important**
   >
   > Dans cette politique, 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 l'accès aux compartiments S3 vers lesquels vous ne prévoyez pas 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 élément `Statement`, remplacez *DOC-EXAMPLE-BUCKET* par le nom d'un compartiment S3 de 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 Actions Amazon S3 dans le Guide de l'utilisateur Amazon Simple Storage Service et dans le billet de blog AWS IAM Policies and Bucket Policies and ACLs! Oh My! (Contrôler l'accès aux ressources S3).

   > **Note**
   >
   > Si vous utilisez plusieurs compartiments, fournissez l'ARN de chacun d'entre eux. Consultez l'exemple suivant pour connaître 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 Suivant : Balises.

7. (Facultatif) Ajoutez des identifications en choisissant Add tag (Ajouter une identification), et en saisissant les identifications préférées de la politique.
8. Choisissez Suivant : vérification.
9. Pour Name (Nom), saisissez un nom pour identifier cette politique, par exemple **`SSMInstanceProfileS3Policy`**.
10. Choisissez Créer une politique.

Considérations en matière de politique supplémentaires pour les instances gérées

Cette section décrit certaines des politiques que vous pouvez ajouter au rôle IAM par défaut créé par la Configuration de gestion des hôtes par défaut, ou les profils d'instance pour AWS Systems Manager. Pour fournir des autorisations de communication entre les instances et l'API Systems Manager, nous vous recommandons de créer des politiques personnalisées reflétant les besoins de votre système et les exigences de sécurité. En fonction de votre plan d'opérations, vous pouvez avoir besoin des autorisations représentées dans une ou plusieurs des trois autres politiques.

Stratégie : **`AmazonSSMDirectoryServiceAccess`**

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

Cette politique AWS gérée permet SSM Agent à AWS Directory Service l'instance gérée d'accéder en votre nom aux demandes de connexion au domaine. Pour plus d'informations, consultez Jonction facile d'une instance Windows EC2 dans le Guide d'administration AWS Directory Service .

Stratégie : **`CloudWatchAgentServerPolicy`**

Obligatoire uniquement si vous prévoyez d'installer et d'exécuter l' CloudWatch agent sur vos instances pour lire les données métriques et de journal d'une instance et les écrire sur Amazon CloudWatch. Ils vous aident à surveiller, à analyser et à répondre rapidement aux problèmes ou aux modifications de vos AWS ressources.

Votre rôle IAM par défaut créé par la configuration de gestion d'hôte par défaut ou le profil d'instance nécessite cette politique uniquement si vous utilisez des fonctionnalités telles qu'Amazon EventBridge ou Amazon CloudWatch Logs. (Vous pouvez également créer une politique plus restrictive qui, par exemple, limite l'accès en écriture à un flux de journal CloudWatch Logs spécifique.)

> **Note**
>
> CloudWatch Les fonctionnalités d'utilisation EventBridge et de journalisation sont facultatives. Toutefois, nous vous recommandons de les paramétrer dès le début de votre processus de configuration de Systems Manager si vous avez décidé de les utiliser. Pour plus d'informations, consultez le guide de EventBridge l'utilisateur Amazon et le guide de l'utilisateur Amazon CloudWatch Logs.

Pour créer des politiques IAM avec des autorisations pour des fonctionnalités Systems Manager supplémentaires, consultez les ressources suivantes :

- Restriction de l'accès aux paramètres Systems Manager à l'aide des politiques IAM
- Configuration d'Automation
- Étape 2 : vérifier ou ajouter des autorisations d'instance pour Session Manager

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

1. Connectez-vous à la console Amazon EC2 AWS Management Console et ouvrez-la 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, sélectionnez Security (Sécurité), Modify IAM role (Modifier le rôle IAM).
5. Pour Rôle IAM, sélectionnez le profil d'instance que vous avez créé via la procédure décrite à l'Configuration alternative pour les autorisations d'instance EC2.
6. Choisissez Update IAM role (Mise à jour du rôle IAM).

Pour de plus amples informations sur l'attachement de rôles IAM à des instances, choisissez l'une des options suivantes, en fonction du type de système d'exploitation sélectionné :

- Associer un rôle IAM à une instance dans le guide de l'utilisateur Amazon EC2
- Associer un rôle IAM à une instance dans le guide de l'utilisateur Amazon EC2

Passez au Améliorez la sécurité des instances EC2 en utilisant des points de terminaison VPC pour Systems Manager.

Améliorez la sécurité des instances EC2 en utilisant des points de terminaison VPC pour Systems Manager

Vous pouvez améliorer le niveau de sécurité de vos nœuds gérés (y compris les machines non EC2 dans un environnement hybride et multicloud) en AWS Systems Manager configurant l'utilisation d'un point de terminaison VPC d'interface dans Amazon Virtual Private Cloud (Amazon VPC). En utilisant un point de terminaison VPC d'interface (point de terminaison d'interface), vous pouvez vous connecter à des services alimentés par. AWS PrivateLink AWS PrivateLink est une technologie qui vous permet d'accéder en privé aux API Amazon Elastic Compute Cloud (Amazon EC2) et Systems Manager en utilisant des adresses IP privées.

AWS PrivateLink restreint tout le trafic réseau entre vos instances gérées, Systems Manager et Amazon EC2 vers le réseau Amazon. Cela signifie que vos instances gérées n'ont pas accès à Internet. Si vous l'utilisez AWS PrivateLink, vous n'avez pas besoin d'une passerelle Internet, d'un périphérique NAT ou d'une passerelle privée virtuelle.

Vous n'êtes pas obligé de le configurer AWS PrivateLink, mais c'est recommandé. Pour plus d'informations sur AWS PrivateLink les points de terminaison VPC, consultez la section et les points de terminaison AWS PrivateLink VPC.

> **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`
>
> L'SSM Agent initie toutes les connexions au service Systems Manager dans le cloud. Vous n'avez donc pas besoin de configurer votre pare-feu pour autoriser le trafic entrant vers vos instances pour Systems Manager.
> Pour de plus amples informations sur ces points de terminaison, consultez Référence : ec2messages, ssmmessages et autres opérations d'API.

À propos d'Amazon VPC

Vous pouvez utiliser Amazon Virtual Private Cloud (Amazon VPC) pour définir un réseau virtuel dans votre propre zone logiquement isolée au sein de ce que l' AWS Cloud on appelle un cloud privé virtuel (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 les instances de votre VPC à internet. Vous pouvez connecter votre VPC à votre propre centre de données d'entreprise, pour en faire AWS Cloud 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, consultez le Guide de l'utilisateur Amazon VPC.

Rubriques

- Limites et restrictions applicables aux points de terminaison de VPC
- Création de points de terminaison de VPC pour Systems Manager
- Création d'une politique de point de terminaison de VPC d'interface

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.

Demandes croisées entre Régions

Les points de terminaison VPC ne prennent pas en charge les demandes interrégionales. Assurez-vous de créer votre point de terminaison au même endroit que votre compartiment. Région AWS Vous pouvez trouver l'emplacement de votre compartiment en utilisant la console Amazon S3 ou la commande get-bucket-location. Utilisez un point de terminaison Amazon S3 spécifique 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, veuillez consulter la rubrique Points de terminaison Amazon S3 dans le Référence générale d'Amazon Web Services. Si vous utilisez le AWS CLI pour envoyer des demandes à Amazon S3, définissez votre région par défaut sur la même région que votre compartiment, ou utilisez le `--region` paramètre 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 Connexions d'appairage de VPC (Quotas) 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 Points de terminaison VPC (Quotas) 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.

Résolution DNS

Si vous utilisez un serveur DNS personnalisé, vous devez ajouter un redirecteur conditionnel pour toutes les requêtes adressées au domaine `amazonaws.com` au serveur Amazon DNS de votre VPC.

Compartiments S3

Votre politique de point de terminaison d'un VPC doit au moins autoriser l'accès aux compartiments Simple Storage Service (Amazon S3) suivants :

- Les compartiments S3 répertoriés dans Communications de l'SSM Agent avec des compartiments S3 gérés par AWS.
- Les compartiments S3 utilisés par Patch Manager pour les opérations d'application de correctifs de base dans votre Région AWS. Ces compartiments contiennent le code qui est extrait et exécuté sur des instances par le service de référence de correctif. Chacun Région AWS possède ses propres compartiments d'opérations de base de correctifs à partir desquels le code est extrait lors de l'exécution d'un document de référence de correctif. Si le code ne peut pas être téléchargé, la commande de référentiel de correctifs échoue.

> **Note**
>
> Si vous utilisez un pare-feu local et que vous prévoyez d'utiliser Patch Manager, ce pare-feu doit également autoriser l'accès au point de terminaison du référentiel de correctifs approprié.

Pour donner accès aux compartiments de votre terminal Région AWS, incluez l'autorisation suivante dans votre politique de point de terminaison.

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

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

Consultez l'exemple suivant.

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

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

Amazon CloudWatch Logs

Si vous n'autorisez pas vos instances à accéder à Internet, créez un point de terminaison VPC pour que les CloudWatch journaux utilisent les fonctionnalités qui envoient des journaux aux CloudWatch journaux. Pour plus d'informations sur la création d'un point de terminaison pour les CloudWatch

journaux, consultez la section Création d'un point de terminaison VPC pour les CloudWatch journaux dans le guide de l'utilisateur Amazon CloudWatch Logs.

Serveur DNS dans un environnement hybride et multicloud

Pour plus d'informations sur la configuration du DNS pour qu'il fonctionne avec des AWS PrivateLink points de terminaison dans des environnements hybrides et multicloud, consultez la section DNS privé pour les points de terminaison d'interface dans le guide de l'utilisateur Amazon VPC. Si vous souhaitez utiliser votre propre DNS, vous pouvez utiliser Route 53 Resolver. Pour plus d'informations, consultez Résolution de requêtes DNS entre des VPC et votre réseau dans le Guide du développeur Amazon Route 53.

Création de points de terminaison de VPC pour Systems Manager

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

Pour créer des points de terminaison de VPC pour Systems Manager

Dans la première étape de cette procédure, 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 Région AWS prise en charge par AWS Systems Manager, par exemple `us-east-2` pour la région USA Est (Ohio). Pour obtenir une liste des valeurs *`region`* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la 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 d'interface suivants :

- **com.amazonaws.*region*.ssm** : point de terminaison pour le service Systems Manager.
- **com.amazonaws.*region*.ec2messages** : Systems Manager utilise ce point de terminaison pour effectuer des appels de SSM Agent au service Systems Manager.
- **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, l'appel pour énumérer les volumes Amazon EBS attachés échoue, ce qui entraîne l'échec de la commande Systems Manager.
- **com.amazonaws.*region*.ssmmessages** : ce point de terminaison est uniquement requis si vous vous connectez à vos instances via un canal de données sécurisé à l'aide de Session Manager. Pour plus d'informations, consultez AWS Systems Manager Session Manager et Référence : ec2messages, ssmmessages et autres opérations d'API.
- **com.amazonaws.*region*.kms** : ce point de terminaison est facultatif. Cependant, il peut être créé si vous souhaitez utiliser le chiffrement AWS Key Management Service (AWS KMS) pour les Parameter Store paramètres Session Manager ou.
- **com.amazonaws.*region*.logs** : ce point de terminaison est facultatif. Toutefois, il peut être créé si vous souhaitez utiliser Amazon CloudWatch Logs (CloudWatch Logs) pour Session ManagerRun Command, ou SSM Agent les journaux.

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 SSM Agent et pour effectuer des opérations d'application de correctifs. Systems Manager utilise également ce point de terminaison pour des tâches telles que charger les journaux de sortie que vous choisissez de stocker dans des compartiments S3, récupérer des scripts ou d'autres fichiers que vous stockez dans des compartiments, etc. Si le groupe de sécurité associé à votre instance restreint le trafic sortant, vous devez ajouter une règle pour autoriser le trafic vers la liste de préfixes pour Amazon S3. Pour plus d'informations, consultez Modifier votre groupe de sécurité dans le Guide AWS PrivateLink .

   Pour plus d'informations sur les compartiments S3 AWS gérés auxquels SSM Agent il est nécessaire d'accéder, consultezCommunications de l'SSM Agent avec des compartiments S3 gérés par AWS. Si vous utilisez un point de terminaison de cloud privé virtuel (VPC) dans vos opérations Systems Manager, vous devez fournir une autorisation explicite dans un profil

d'instance EC2 pour Systems Manager, ou dans une fonction du service pour les nœuds gérés non EC2 d'un environnement hybride et multicloud.

Création d'une politique de point de terminaison de VPC d'interface

Vous pouvez créer des politiques pour les points de terminaison de l'interface VPC AWS Systems Manager dans lesquelles vous pouvez spécifier :

- Le principal qui peut exécuter des actions.
- Les actions qui peuvent être effectuées.
- Les ressources qui peuvent avoir des actions exécutées sur elles.

Pour plus d'informations, consultez Contrôle de l'accès aux services avec points de terminaison d'un VPC dans le Guide de l'utilisateur Amazon VPC.

Utilisation de Systems Manager dans des environnements hybrides et multicloud

Vous pouvez l'utiliser AWS Systems Manager pour gérer à la fois des instances Amazon Elastic Compute Cloud (EC2) et un certain nombre de types de machines non EC2. Cette section décrit les tâches de configuration effectuées par les administrateurs système et de compte pour gérer des machines non EC2 à l'aide de Systems Manager dans un environnement hybride et multicloud. Une fois ces étapes terminées, les utilisateurs auxquels l' Compte AWS administrateur a accordé des autorisations peuvent utiliser Systems Manager pour configurer et gérer les machines non EC2 de leur organisation.

Toute machine configurée pour être utilisée avec Systems Manager est un nœud géré.

> **Note**
>
> - Vous pouvez enregistrer des appareils de périphérie en tant que nœuds gérés en utilisant les mêmes étapes d'activation hybride que celles utilisées pour d'autres machines non EC2. Ces types de périphériques périphériques incluent à la fois les AWS IoT appareils et les appareils autres que AWS IoT les appareils. Utilisez le processus décrit dans cette section pour configurer ces types d'appareils de périphérie.

Systems Manager prend également en charge les périphériques utilisant le logiciel AWS IoT Greengrass Core. Le processus de configuration et les exigences pour les périphériques AWS IoT Greengrass principaux sont différents de ceux pour AWS IoT les périphériques périphériques autres que les AWS périphériques périphériques. Pour plus d'informations sur l'enregistrement AWS IoT Greengrass des appareils en vue de leur utilisation avec Systems Manager, consultezGestion des appareils de pointe avec Systems Manager.

- Les machines macOS non EC2 ne sont pas prises en charge pour les environnements hybrides et multicloud Systems Manager.

Si vous prévoyez d'utiliser Systems Manager pour gérer des instances Amazon Elastic Compute Cloud (Amazon EC2), ou d'utiliser des instances Amazon EC2 et des machines non EC2 dans un environnement hybride et multicloud, commencez par suivre les étapes de la section Utilisation de Systems Manager avec des instances EC2.

Après avoir configuré votre environnement hybride et multicloud pour Systems Manager, vous pourrez effectuer les opérations suivantes :

- Créer une procédure cohérente et sécurisée pour gérer à distance vos charges de travail hybrides et multicloud 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 machines à l'aide de AWS Identity and Access Management (IAM).
- Centralisez l'audit des opérations effectuées sur vos machines en consultant l'activité de l'API enregistrée dans AWS CloudTrail.

  Pour plus d'informations sur l'utilisation CloudTrail pour surveiller les actions de Systems Manager, consultezJournalisation des appels d' AWS Systems Manager API avec AWS CloudTrail.
- Centralisez la surveillance en configurant Amazon EventBridge et Amazon Simple Notification Service (Amazon SNS) pour envoyer des notifications concernant le succès de l'exécution du service.

  Pour plus d'informations sur l'utilisation EventBridge pour surveiller les événements de Systems Manager, consultezSurveillance d'événements Systems Manager avec Amazon EventBridge.

À propos des nœuds gérés

Une fois que vous avez terminé de configurer vos machines non-EC2 pour Systems Manager comme décrit dans cette section, vos machines activées par des hybrides sont répertoriées AWS Management Console et décrites comme des nœuds gérés. Dans la console, néanmoins, les ID de vos nœuds gérés activés par un système hybride se distinguent des instances Amazon EC2 par le préfixe « mi- ». Les ID d'instance Amazon EC2 utilisent le préfixe « i- ».

Un nœud géré est une machine configurée pour Systems Manager. Auparavant, les nœuds gérés étaient tous appelés instances gérées. Le terme instance fait désormais référence uniquement aux instances EC2. Les commandes deregister-managed-instance ont été nommées avant ce changement de terminologie.

Pour plus d'informations, consultez Utilisation de nœuds gérés.

À propos des niveaux d'instances

Systems Manager offre un niveau d'instances standard et un niveau d'instances avancées pour les nœuds gérés non EC2 de votre environnement hybride et multicloud. Le niveau d'instances standard vous permet d'enregistrer un maximum de 1 000 machines activées par un système hybride par Compte AWS et par Région AWS. Si vous avez besoin d'enregistrer plus de 1 000 machines non EC2 dans un seul compte et une seule région, utilisez le niveau d'instances avancées. Les instances avancées vous permettent également de vous connecter à vos machines non-EC2 en utilisant. AWS Systems Manager Session Manager Session Managerfournit un accès shell interactif à vos nœuds gérés.

Pour plus d'informations, voir Configuration des niveaux d'instance.

Rubriques

- Créez le rôle de service IAM requis pour Systems Manager dans les environnements hybrides et multicloud
- Créez une activation hybride pour enregistrer les nœuds auprès de Systems Manager
- Comment installer le SSM Agent sur des nœuds Linux hybrides
- Comment installer le SSM Agent sur des Windows nœuds hybrides

Créez le rôle de service IAM requis pour Systems Manager dans les environnements hybrides et multicloud

Les machines non EC2 (Amazon Elastic Compute Cloud) dans un environnement hybride et multicloud nécessitent un rôle de service AWS Identity and Access Management (IAM) pour

communiquer avec le service. AWS Systems Manager Le rôle octroie à AWS Security Token Service (AWS STS) l'approbation `AssumeRole` au service Systems Manager. Vous devez uniquement créer une fonction du service pour un environnement hybride et multicloud une fois pour chaque Compte AWS. Toutefois, vous pouvez choisir de créer plusieurs fonctions du service pour différentes activations hybrides si les machines de votre environnement hybride et multicloud requièrent des autorisations différentes.

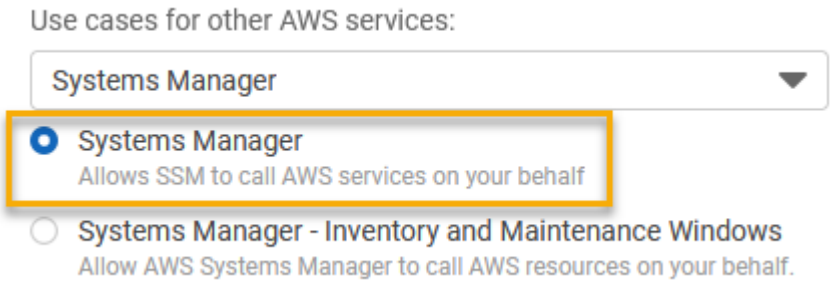
Les procédures suivantes expliquent comment créer la fonction de service requise à l'aide de la console Systems Manager ou de votre outil de ligne de commande préféré.

Utilisation du AWS Management Console pour créer un rôle de service IAM pour les activations hybrides de Systems Manager

Utilisez la procédure suivante pour créer une fonction de service pour une activation hybride. Cette procédure utilise la politique `AmazonSSMManagedInstanceCore` pour la fonctionnalité principale de Systems Manager. Selon votre cas d'utilisation, vous devrez peut-être ajouter des politiques supplémentaires à votre fonction de service pour que vos machines sur site puissent accéder à d'autres fonctionnalités ou Services AWS. Par exemple, sans accès aux éléments requis aux compartiments gérés AWS Amazon Simple Storage Service (Amazon S3), les opérations de correctifs Patch Manager échouent.

Pour créer un rôle de service (console)

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le volet de navigation, sélectionnez Rôles, puis Créer un rôle.
3. Pour Select trusted entity (Sélectionner une entité de confiance), effectuez les choix suivants :
    1. Pour Trusted entity (Entité de confiance), choisissez Service AWS.
    2. Pour les autres cas d'utilisation Services AWS, choisissez Systems Manager.
    3. Choisissez Systems Manager, comme illustré dans l'image suivante.



4. Choisissez Suivant.
5. Sur la page Add permissions (Ajouter des autorisations), procédez comme suit :
   - Utilisez le champ de recherche pour trouver la politique ManagedInstanceprincipale d'AmazonSSM. Cochez la case en regard de son nom.

   - La console conserve votre sélection même si vous recherchez d'autres politiques.
   - Si vous avez créé une politique de compartiment S3 personnalisée au cours de la procédure (Facultatif) créer une politique personnalisée pour l'accès au compartiment S3, recherchez-la et cochez la case en regard de son nom.
   - Si vous envisagez de joindre des machines non EC2 à un Active Directory géré par AWS Directory Service, recherchez AmazonSSM DirectoryService Access et cochez la case à côté de son nom.
   - Si vous prévoyez d'utiliser EventBridge ou CloudWatch Logs pour gérer ou surveiller votre nœud géré, recherchez CloudWatchAgentServerPolicy et cochez la case à côté de son nom.
6. Choisissez Suivant.
7. Pour Nom de rôle, saisissez un nom pour votre nouveau rôle de serveur IAM, par exemple, **`SSMServerRole`**.

> **Note**
>
> Notez le nom de rôle. Vous pouvez choisir ce rôle lorsque vous enregistrez de nouvelles machines à gérer à l'aide de Systems Manager.

8. (Facultatif) Pour Description, mettez à jour la description pour ce rôle de serveur IAM.
9. (Facultatif) Pour Tags (Balises), ajoutez une ou plusieurs paires clé-valeur de balise afin d'organiser, de suivre ou de contrôler l'accès pour ce rôle.
10. Sélectionnez Créer un rôle. Le système vous renvoie à la page Rôles.

Utilisation du AWS CLI pour créer un rôle de service IAM pour les activations hybrides de Systems Manager

Utilisez la procédure suivante pour créer une fonction de service pour une activation hybride. Cette procédure utilise la politique `AmazonSSMManagedInstanceCore` pour la fonctionnalité principale de Systems Manager. Selon votre cas d'utilisation, vous devrez peut-être ajouter des politiques supplémentaires à votre fonction du service pour vos machines non EC2 dans un environnement hybride et multicloud afin de pouvoir accéder à d'autres capacités ou Services AWS.

Exigence pour les politiques de compartiment S3

Dans les cas suivants, vous devez créer une politique d'autorisation IAM personnalisée pour les compartiments Amazon Simple Storage Service (Amazon S3) avant de terminer cette procédure :

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

AWS CLI

Pour créer une fonction du service IAM pour un environnement hybride et multicloud (AWS CLI)

1. Installez et configurez le AWS Command Line Interface (AWS CLI), si ce n'est pas déjà fait.

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l' AWS CLI.
2. Sur votre machine locale, créez un fichier texte avec un nom tel que `SSMService-Trust.json` avec la politique d'approbation suivante. Assurez-vous d'enregistrer le fichier avec l'extension de fichier `.json`. Assurez-vous de spécifier votre Compte AWS et le Région AWS dans l'ARN dans lequel vous avez créé votre activation hybride.

   ```
   {
   ```

```
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"",
         "Effect":"Allow",
         "Principal":{
            "Service":"ssm.amazonaws.com"
         },
         "Action":"sts:AssumeRole",
         "Condition":{
            "StringEquals":{
               "aws:SourceAccount":"123456789012"
            },
            "ArnEquals":{
               "aws:SourceArn":"arn:aws:ssm:us-east-2:123456789012:*"
            }
         }
      }
   ]
}
```

3. Ouvrez le AWS CLI, et dans le répertoire où vous avez créé le fichier JSON, exécutez 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 & macOS

   ```
   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
   ```

4. Exécutez la commande 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 nœud géré à exécuter des commandes à l'aide de Systems Manager.

> **Note**
>
> Les politiques que vous ajoutez pour un profil de service pour des nœuds gérés dans un environnement hybride et multicloud sont les mêmes politiques que celles utilisées pour créer un profil d'instance pour des instances Amazon Elastic Compute Cloud (Amazon EC2). Pour plus d'informations sur les AWS politiques utilisées dans les commandes suivantes, consultez Configurer les autorisations d'instance requises pour Systems Manager.

(Obligatoire) Exécutez la commande suivante pour autoriser un nœud géré à utiliser les fonctionnalités AWS Systems Manager de base du service.

Linux & macOS

```
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 politique de compartiment S3 personnalisée pour votre rôle de service, exécutez la commande suivante pour autoriser AWS Systems Manager Agent (SSM Agent) à accéder aux compartiments que vous avez spécifiés dans la politique. Remplacez *account-id* et *DOC-EXAMPLE-BUCKET par votre ID et le nom de votre Compte AWS bucket*.

Linux & macOS

```
aws iam attach-role-policy \
    --role-name SSMServiceRole \
    --policy-arn arn:aws:iam::account-id:policy/DOC-EXAMPLE-BUCKET
```

Windows

```
aws iam attach-role-policy ^
    --role-name SSMServiceRole ^
    --policy-arn arn:aws:iam::account-id:policy/DOC-EXAMPLE-BUCKET
```

(Facultatif) Exécutez la commande suivante SSM Agent pour autoriser AWS Directory Service le nœud géré à accéder en votre nom aux demandes de connexion au domaine. Votre fonction du service requiert uniquement cette politique si vous joignez vos nœuds à un annuaire Microsoft AD.

Linux & macOS

```
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 autoriser l' CloudWatch agent à s'exécuter sur vos nœuds gérés. Cette commande permet de lire des informations sur un nœud et de les y écrire CloudWatch. Votre profil de service n'a besoin de cette politique que si vous utilisez des services tels qu'Amazon EventBridge ou Amazon CloudWatch Logs.

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

Tools for PowerShell

Pour créer une fonction du service IAM pour un environnement hybride et multicloud (AWS Tools for Windows PowerShell)

1. Installez et configurez les AWS Tools for PowerShell (Outils pour Windows PowerShell), si ce n'est pas déjà fait.

   Pour plus d'informations, consultez Installation d' AWS Tools for PowerShell.

2. Sur votre machine locale, créez un fichier texte avec un nom tel que `SSMService-Trust.json` avec la politique d'approbation suivante. Assurez-vous d'enregistrer le fichier avec l'extension de fichier `.json`. Assurez-vous de spécifier votre Compte AWS et le Région AWS dans l'ARN dans lequel vous avez créé votre activation hybride.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"",
         "Effect":"Allow",
         "Principal":{
            "Service":"ssm.amazonaws.com"
         },
         "Action":"sts:AssumeRole",
         "Condition":{
            "StringEquals":{
               "aws:SourceAccount":"123456789012"
            },
            "ArnEquals":{
               "aws:SourceArn":"arn:aws:ssm:region:123456789012:*"
            }
         }
      }
   ]
}
```

3. Ouvrez PowerShell en mode administratif, et dans le répertoire où vous avez créé le fichier JSON, exécutez 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)
```

4. Utilisez Register-IAM RolePolicy comme suit pour autoriser le rôle de service que vous avez créé à créer un jeton de session. Ce jeton de session autorise votre nœud géré à exécuter des commandes à l'aide de Systems Manager.

> **Note**
>
> Les politiques que vous ajoutez pour un profil de service pour des nœuds gérés dans un environnement hybride et multicloud sont les mêmes politiques que celles utilisées pour créer un profil d'instance pour des instances EC2. Pour plus d'informations sur les AWS politiques utilisées dans les commandes suivantes, consultez Configurer les autorisations d'instance requises pour Systems Manager.

(Obligatoire) Exécutez la commande suivante pour autoriser un nœud géré à utiliser les fonctionnalités AWS Systems Manager de base du service.

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

Si vous avez créé une politique de compartiment S3 personnalisée pour votre rôle de service, exécutez la commande suivante pour permettre à SSM Agent d'accéder aux compartiments que vous avez spécifiés dans la politique. Remplacez *account-id* et *my-bucket-policy-name* par l'ID de votre Compte AWS 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 SSM Agent pour autoriser AWS Directory Service le nœud géré à accéder en votre nom aux demandes de connexion au domaine. Votre rôle de serveur requiert uniquement cette politique si vous joignez vos nœuds à un annuaire Microsoft AD.

```
Register-IAMRolePolicy `
    -RoleName SSMServiceRole `
```

```
    -PolicyArn arn:aws:iam::aws:policy/AmazonSSMDirectoryServiceAccess
```

(Facultatif) Exécutez la commande suivante pour autoriser l' CloudWatch agent à s'exécuter sur vos nœuds gérés. Cette commande permet de lire des informations sur un nœud et de les y écrire CloudWatch. Votre profil de service n'a besoin de cette politique que si vous utilisez des services tels qu'Amazon EventBridge ou Amazon CloudWatch Logs.

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

Passez au Créez une activation hybride pour enregistrer les nœuds auprès de Systems Manager.

Créez une activation hybride pour enregistrer les nœuds auprès de Systems Manager

Pour configurer des machines autres que des instances Amazon Elastic Compute Cloud (EC2) en tant que nœuds gérés pour un environnement hybride et multicloud, vous devez créer et appliquer une activation hybride. Une fois que l'activation a abouti, vous recevez immédiatement un code d'activation et un ID d'activation en haut de la page de la console. Vous spécifiez cette combinaison de code et d'identifiant lors de l'installation AWS Systems Manager SSM Agent sur des machines autres que EC2 pour votre environnement hybride et multicloud. La combinaison code/ID fournit un accès sécurisé au service Systems Manager à partir de vos nœuds gérés.

> **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 avec Systems Manager. Une activation expirée n'a aucun impact sur les serveurs et VM que vous avez préalablement enregistrés avec Systems Manager. Si une activation expire,

vous ne pouvez pas enregistrer d'autres serveurs et machines virtuelles avec Systems Manager à l'aide de cette activation spécifique. Il vous suffit d'en créer une.

Chaque serveur et chaque VM sur site que vous avez précédemment enregistré(e) reste enregistré(e) comme nœud géré par Systems Manager tant que vous n'aurez pas annulé son enregistrement de manière explicite. Vous pouvez désenregistrer un nœud géré Fleet Manager dans l'onglet Managed nodes de la console Systems Manager à l'aide de la AWS CLI commande deregister-managed-instanceou de l'appel d'API. DeregisterManagedInstance

À propos des nœuds gérés

Un nœud géré est une machine configurée pour AWS Systems Manager. AWS Systems Manager prend en charge les instances Amazon Elastic Compute Cloud (Amazon EC2), les appareils périphériques et les serveurs ou machines virtuelles sur site, y compris les machines virtuelles d'autres environnements cloud. Auparavant, les nœuds gérés étaient tous appelés instances gérées. Le terme instance fait désormais référence uniquement aux instances EC2. Les commandes deregister-managed-instance ont été nommées avant ce changement de terminologie.

À propos des balises d'activation

Si vous créez une activation en utilisant le AWS Command Line Interface (AWS CLI) ou AWS Tools for 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 AWS CLI exemple de commande à exécuter sur une machine Linux locale qui inclut des balises facultatives.

```
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 nœuds gérés lorsque vous les activez.

Vous ne pouvez pas ajouter ou supprimer des balises dans une activation existante. Si vous ne souhaitez pas affecter automatiquement des balises à vos serveurs et machines virtuelles sur site à l'aide d'une activation, vous pouvez leur ajouter des balises ultérieurement. Plus particulièrement,

vous pouvez baliser vos serveurs et machines virtuelles sur site une fois qu'ils se sont connectés à Systems Manager pour la première fois. Une fois qu'ils se sont connectés, un ID de nœud géré leur est affecté et ils sont répertoriés dans la console Systems Manager avec un ID dont le préfixe est « mi- ». Pour plus d'informations sur la façon d'ajouter des balises à vos nœuds gérés sans utiliser le processus d'activation, consultez la rubrique Balisage des nœuds gérés.

> **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 le créer à l'aide du AWS CLI ou des 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 Annulation de l'enregistrement de nœuds gérés dans un environnement hybride et multicloud.

Rubriques

- Utilisation du AWS Management Console pour créer une activation permettant d'enregistrer des nœuds gérés auprès de Systems Manager
- Utilisation de la ligne de commande pour créer une activation permettant d'enregistrer les nœuds gérés auprès de Systems Manager

Utilisation du AWS Management Console pour créer une activation permettant d'enregistrer des nœuds gérés auprès de Systems Manager

Créer une activation de nœuds gérés

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

   -ou-

   Si vous accédez aux activations hybrides pour la première fois actuellement Région AWS, choisissez Créer une activation.

4. (Facultatif) Dans le champ Activation description (Description de l'activation), saisissez une description pour cette activation. Nous vous recommandons de saisir une description si vous prévoyez d'activer un grand nombre de serveurs et de machines virutelles.
5. Pour Limite d'instances, spécifiez le nombre total de nœuds auprès desquels vous souhaitez vous enregistrer dans AWS le cadre de cette activation. La valeur par défaut est 1 instance.
6. Pour le rôle IAM, choisissez une option de rôle de service qui permet à vos serveurs et machines virtuelles de communiquer AWS Systems Manager dans le cloud :

   - Option 1 : choisissez Use the default role created by the system (Utiliser le rôle par défaut créé par le système) pour utiliser un rôle et une politique gérée fournis par AWS.
   - Option 2 : choisissez Select an existing custom IAM role that has the required permissions (Sélectionner un rôle IAM personnalisé existant ayant les autorisations requises) pour utiliser le rôle personnalisé facultatif que vous avez créé précédemment. Ce rôle doit avoir une politique de relation d'approbation qui spécifie `"Service": "ssm.amazonaws.com"`. Si votre rôle IAM ne spécifie pas ce principe dans une politique de relation d'approbation, l'erreur suivante s'affiche :

     ```
     An error occurred (ValidationException) when calling the CreateActivation
                                        operation: Not existing role:
      arn:aws:iam::<accountid>:role/SSMRole
     ```

     Pour plus d'informations sur la création de ce rôle, consultez la page Créez le rôle de service IAM requis pour Systems Manager dans les environnements hybrides et multicloud.
7. Dans le champ Activation expiry date (Date d'expiration de l'activation), indiquez une date d'expiration pour l'activation. La date d'expiration doit se situer dans la plage des 30 prochains jours. La valeur par défaut est 24 heures.

   > **Note**
   >
   > Si vous souhaitez enregistrer des nœuds gérés supplémentaires après la date d'expiration, vous devez créer une nouvelle activation. La date d'expiration n'a aucun impact sur les nœuds enregistrés et en cours d'exécution.

8. (Facultatif) Pour le champ Default instance name (Nom de l'instance par défaut), spécifiez une valeur de nom d'identification à afficher pour tous les nœuds gérés associés à cette activation.
9. Choisissez Créer une activation. Systems Manager renvoie immédiatement le code d'activation et l'ID à la console.

Utilisation de la ligne de commande pour créer une activation permettant d'enregistrer les nœuds gérés auprès de Systems Manager

La procédure suivante décrit comment utiliser le AWS Command Line Interface (AWS CLI) (sous Linux ouWindows) ou comment AWS Tools for PowerShell créer une activation de nœud géré.

Pour créer une activation

1. Installez et configurez le AWS CLI ou le AWS Tools for PowerShell, si ce n'est pas déjà fait.

   Pour plus d'informations, consultez la section Installation ou mise à jour de la version la plus récente de l' AWS CLI et Installation d' AWS Tools for PowerShell.
2. Exécutez la commande suivante pour créer une activation.

> **Note**
>
> - Dans la commande ci-après, remplacez *`region`* (région) par vos propres informations. Pour obtenir une liste des valeurs *`region`* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la Référence générale d'Amazon Web Services.
> - Le rôle que vous spécifiez pour le paramètre *`iam-role (rôle IAM)`* doit avoir une politique de relation d'approbation qui spécifie `"Service": "ssm.amazonaws.com"`. Si votre rôle AWS Identity and Access Management (IAM) ne spécifie pas ce principe dans une politique de relation de confiance, le message d'erreur suivant s'affiche :
>
>   ```
>   An error occurred (ValidationException) when calling the CreateActivation
>                                           operation: Not existing role:
>    arn:aws:iam::<accountid>:role/SSMRole
>   ```
>
>   Pour plus d'informations sur la création de ce rôle, consultez la page Créez le rôle de service IAM requis pour Systems Manager dans les environnements hybrides et multicloud.
> - Pour `--expiration-date`, fournissez une date au format horodatage, `"2021-07-07T00:00:00"` par exemple, pour indiquer la date d'expiration du code d'activation. Vous pouvez spécifier une date jusqu'à 30 jours à l'avance. Si vous ne fournissez pas de date d'expiration, le code d'activation expire sous 24 heures.

Linux & macOS

```
aws ssm create-activation \
    --default-instance-name name \
    --iam-role iam-service-role-name \
    --registration-limit number-of-managed-instances \
    --region region \
    --expiration-date "timestamp" \\
    --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 ^
    --expiration-date "timestamp" ^
    --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 `
    -ExpirationDate "timestamp" `
    -Tag @{"Key"="key-name-1";"Value"="key-value-1"},@{"Key"="key-
name-2";"Value"="key-value-2"}
```

Voici un exemple.

Linux & macOS

```
aws ssm create-activation \
    --default-instance-name MyWebServers \
    --iam-role service-role/AmazonEC2RunCommandRoleForManagedInstances \
    --registration-limit 10 \
    --region us-east-2 \
```

```
    --expiration-date "2021-07-07T00:00:00" \
    --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 ^
    --expiration-date "2021-07-07T00:00:00" ^
    --tags "Key=Environment,Value=Production" "Key=Department,Value=Finance"
```

PowerShell

```
New-SSMActivation -DefaultInstanceName MyWebServers `
    -IamRole service-role/AmazonEC2RunCommandRoleForManagedInstances `
    -RegistrationLimit 10 `
    -Region us-east-2 `
    -ExpirationDate "2021-07-07T00:00:00" `
    -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.

Comment installer le SSM Agent sur des nœuds Linux hybrides

Cette rubrique décrit comment procéder à l'installation AWS Systems Manager SSM Agent sur des machines Linux autres que EC2 (Amazon Elastic Compute Cloud) dans un environnement hybride et multicloud. Si vous prévoyez d'utiliser des machines Windows Server dans un environnement hybride et multicloud, consultez l'étape suivante, Comment installer le SSM Agent sur des Windows nœuds hybrides.

> **Important**
>
> Cette procédure concerne des types de machines autres que les instances EC2 pour un environnement hybride et multicloud. Pour télécharger et installer l'SSM Agent sur une

instance EC2 pour Linux, reportez-vous à la section Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Linux.

Avant de commencer, recherchez le code d'activation et l'ID d'activation qui vous ont été envoyés à la fin de l'activation hybride, plus haut dans la rubrique Créez une activation hybride pour enregistrer les nœuds auprès de Systems Manager. Vous indiquez le code et l'ID dans la procédure suivante.

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

Par exemple, pour télécharger SSM Agent pour Amazon Linux, RHEL, CentOS et SLES 64 bits à partir de la région USA Est (Ohio) (us-east-2), utilisez l'URL suivante :

```
https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/linux_amd64/amazon-ssm-agent.rpm
```

Amazon Linux 1,Amazon Linux 2, RHEL, Oracle Linux, CentOS, and SLES

- x86_64

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

- x86

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

- ARM64

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

RHEL 6.x, CentOS 6.x

- x86_64

```
https://s3.region.amazonaws.com/amazon-ssm-region/3.0.1479.0/
linux_amd64/amazon-ssm-agent.rpm
```

- x86

```
https://s3.region.amazonaws.com/amazon-ssm-region/3.0.1479.0/
linux_386/amazon-ssm-agent.rpm
```

Ubuntu Server

- x86_64

```
https://s3.region.amazonaws.com/amazon-ssm-region/latest/debian_amd64/
amazon-ssm-agent.deb
```

- ARM64

```
https://s3.region.amazonaws.com/amazon-ssm-region/latest/debian_arm64/
amazon-ssm-agent.deb
```

- x86

```
https://s3.region.amazonaws.com/amazon-ssm-region/latest/debian_386/
amazon-ssm-agent.deb
```

Debian Server

- x86_64

```
https://s3.region.amazonaws.com/amazon-ssm-region/latest/debian_amd64/
amazon-ssm-agent.deb
```

- ARM64

```
https://s3.region.amazonaws.com/amazon-ssm-region/latest/debian_arm64/
amazon-ssm-agent.deb
```

Raspberry Pi OS (formerly Raspbian)

- `https://s3.region.amazonaws.com/amazon-ssm-region/latest/debian_arm/amazon-ssm-agent.deb`

Pour installer SSM Agent sur des machines non EC2 dans un environnement hybride et multicloud

1. Connectez-vous à un serveur ou une VM de votre environnement hybride et multicloud.
2. Si vous utilisez un proxy HTTP ou HTTPS, vous devez définir les variables d'environnement `http_proxy` ou `https_proxy` dans la session shell en cours. Si vous n'utilisez pas de proxy, vous pouvez ignorer cette étape.

   Pour un serveur proxy HTTP, saisissez les commandes suivantes sur la ligne de commande :

   ```
   export http_proxy=http://hostname:port
   export https_proxy=http://hostname:port
   ```

   Pour un serveur proxy HTTPS, saisissez les commandes suivantes sur la ligne de commande :

   ```
   export http_proxy=http://hostname:port
   export https_proxy=https://hostname:port
   ```

3. 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 lors de la création d'une activation de nœuds gérés, et par l'identifiant de la Région AWS depuis laquelle vous souhaitez télécharger SSM Agent, puis appuyez sur `Enter`.

   > **Note**
   >
   > Prenez note des informations importantes suivantes :
   >
   > - `sudo` n'est pas nécessaire si vous êtes un utilisateur racine.
   > - Téléchargez `ssm-setup-cli` à partir de Région AWS l'endroit où votre activation hybride a été créée.
   > - `ssm-setup-cli` prend en charge une option `manifest-url` qui détermine la source à partir de laquelle l'agent est téléchargé. Ne spécifiez aucune valeur pour cette option à moins que votre organisation ne l'exige.
   > - Lors de l'enregistrement des instances, n'utilisez que le lien de téléchargement fourni pour `ssm-setup-cli`. `ssm-setup-cli` ne doit pas être stocké séparément pour une utilisation ultérieure.
   > - Vous pouvez utiliser le script fourni ici pour valider la signature de`ssm-setup-cli`.

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

De plus, `ssm-setup-cli` inclut les options suivantes :

- `version` : les valeurs valides sont `latest` et `stable`.
- `downgrade` : permet de rétrograder SSM Agent à une version antérieure. Spécifiez `true` pour installer une version antérieure de l'agent.
- `skip-signature-validation` : ignore la validation de signature lors du téléchargement et de l'installation de l'agent.

RHEL 6.x et CentOS 6.x

```
mkdir /tmp/ssm
curl https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/3.0.1479.0/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 -E amazon-ssm-agent -register -code "activation-code" -id "activation-id" -region
 "region"
sudo start amazon-ssm-agent
```

Amazon Linux 1

```
mkdir /tmp/ssm
curl https://amazon-ssm-region.s3.region.amazonaws.com/latest/linux_amd64/ssm-setup-cli
 -o /tmp/ssm/ssm-setup-cli
sudo chmod +x /tmp/ssm/ssm-setup-cli
sudo /tmp/ssm/ssm-setup-cli -register -activation-code "activation-code" -id
 "activation-id" -region "region"
```

Amazon Linux 2, RHEL 7.xOracle Linux, CentOS 7.x et SLES

```
mkdir /tmp/ssm
curl https://amazon-ssm-region.s3.region.amazonaws.com/latest/linux_amd64/ssm-setup-cli
 -o /tmp/ssm/ssm-setup-cli
```

```
sudo chmod +x /tmp/ssm/ssm-setup-cli
sudo /tmp/ssm/ssm-setup-cli -register -activation-code "activation-code" -activation-id
 "activation-id" -region "region"
```

RHEL 8.x et CentOS 8.x

```
mkdir /tmp/ssm
curl https://amazon-ssm-region.s3.region.amazonaws.com/latest/linux_amd64/ssm-setup-cli
 -o /tmp/ssm/ssm-setup-cli
sudo chmod +x /tmp/ssm/ssm-setup-cli
sudo /tmp/ssm/ssm-setup-cli -register -activation-code "activation-code" -activation-id
 "activation-id" -region "region"
```

Debian Server

```
mkdir /tmp/ssm
curl https://amazon-ssm-region.s3.region.amazonaws.com/latest/debian_amd64/ssm-setup-
cli -o /tmp/ssm/ssm-setup-cli
sudo chmod +x /tmp/ssm/ssm-setup-cli
sudo /tmp/ssm/ssm-setup-cli -register -activation-code "activation-code" -activation-id
 "activation-id" -region "region"
```

Raspberry Pi OS (anciennement Raspbian)

```
mkdir /tmp/ssm
curl https://amazon-ssm-region.s3.region.amazonaws.com/latest/debian_arm/ssm-setup-cli
 -o /tmp/ssm/ssm-setup-cli
sudo chmod +x /tmp/ssm/ssm-setup-cli
sudo /tmp/ssm/ssm-setup-cli -register -activation-code "activation-code" -activation-id
 "activation-id" -region "region"
```

Ubuntu

- Utilisation de packages .deb

  ```
  mkdir /tmp/ssm
  curl https://amazon-ssm-region.s3.region.amazonaws.com/latest/debian_amd64/ssm-setup-
  cli -o /tmp/ssm/ssm-setup-cli
  sudo chmod +x /tmp/ssm/ssm-setup-cli
  sudo /tmp/ssm/ssm-setup-cli -register -activation-code "activation-code" -activation-
  id "activation-id" -region "region"
  ```

- 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.

  Sur Ubuntu Server 20.10 STR & 20.04, 18.04 et 16.04 LTS, les fichiers du programme d'installation SSM Agent, 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 répertoire `/snap` vers le répertoire `/etc/amazon/ssm/`. Les fichiers journaux et de bibliothèque n'ont pas changé (`/var/lib/amazon/ssm`, `/var/log/amazon/ssm`).

  ```
  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 de Snaps contient la dernière version de l'SSM Agent ; pas le canal stable. Si vous souhaitez suivre les informations de version de SSM Agent sur le canal candidat, exécutez la commande suivante sur vos nœuds gérés 64 bits Ubuntu Server 18.04 et 16.04 LTS.

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

La commande télécharge et installe SSM Agent sur la machine activée par un système hybride dans votre environnement hybride et multicloud. La commande arrête SSM Agent, puis enregistre la machine auprès du service Systems Manager. La machine est désormais un nœud géré. Les instances Amazon EC2 configurées pour Systems Manager sont également des nœuds gérés. Pourtant, dans la console Systems Manager, vos nœuds activés par un système hybride se distinguent des instances Amazon EC2 par le préfixe « mi- ».

Passez au Comment installer le SSM Agent sur des Windows nœuds hybrides.

Configuration de la rotation automatique de la clé privée

Pour renforcer votre niveau de sécurité, vous pouvez configurer AWS Systems Manager Agent (SSM Agent) pour qu'il fasse automatiquement pivoter la clé privée pour votre environnement hybride et multicloud. Vous pouvez accéder à cette fonction en utilisant la version 3.0.1031.0 ou ultérieure de l'SSM Agent. Procédez comme suit pour activer cette fonction.

Pour configurer SSM Agent de sorte à effectuer une rotation de la clé privée d'un environnement hybride et multicloud

1. Accédez à `/etc/amazon/ssm/` sur une machine Linux ou à `C:\Program Files\Amazon \SSM` sur une machine Windows.
2. Copiez le contenu de `amazon-ssm-agent.json.template` vers un nouveau fichier appelé `amazon-ssm-agent.json`. Enregistrez `amazon-ssm-agent.json` dans le même répertoire que `amazon-ssm-agent.json.template`.
3. Recherchez `Profile`, `KeyAutoRotateDays`. Saisissez le nombre de jours qui doit séparer les rotations automatiques de clé privée.
4. Redémarrez SSM Agent.

Chaque fois que vous modifiez la configuration, redémarrez l'SSM Agent.

Vous pouvez personnaliser d'autres fonctions de l'SSM Agent en suivant la même procédure. Pour une up-to-date liste des propriétés de configuration disponibles et de leurs valeurs par défaut, consultez la section Définitions des propriétés de configuration.

Désenregistrer et réenregistrer un nœud géré

Vous pouvez annuler l'enregistrement d'un nœud géré activé de manière hybride en appelant l'opération DeregisterManagedInstance API depuis Windows AWS CLI ou depuis Tools for Windows. PowerShell Voici un exemple de commande de l'interface de ligne de commande :

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

Pour supprimer les informations d'enregistrement restantes pour l'agent, supprimez la clé `IdentityConsumptionOrder` du fichier `amazon-ssm-agent.json`. Ensuite, exécutez la commande suivante :

```
amazon-ssm-agent -register -clear
```

Vous pouvez réenregistrer une machine après avoir annulé son enregistrement. Procédez comme suit pour réenregistrer une machine. Une fois la procédure terminée, votre nœud géré s'affiche à nouveau dans la liste des nœuds gérés.

Pour réenregistrer un nœud géré sur une machine Linux non EC2

1. Connectez-vous à votre machine.
2. Exécutez la commande suivante. Veillez à remplacer les valeurs d'espace réservé par le code d'activation et l'ID d'activation générés lors de la création d'une activation de nœuds gérés, et par l'identifiant de la région depuis laquelle vous souhaitez télécharger SSM Agent.

```
echo "yes" | sudo /tmp/ssm/ssm-setup-cli -register -activation-code "activation-code" -activation-id "activation-id" -region "region
```

Résolution des problèmes d'installation de SSM Agent sur des machines Linux non EC2

Utilisez les informations suivantes pour résoudre les problèmes liés à l'installation de SSM Agent sur des machines Linux activées par un système hybride dans un environnement hybride et multicloud.

Vous recevez un DeliveryTimedOut message d'erreur

Problème : lors de la configuration d'une machine en Compte AWS tant que nœud géré pour une machine séparée Compte AWS, vous recevez `DeliveryTimedOut` après avoir exécuté les commandes d'installation SSM Agent sur la machine cible.

Solution : `DeliveryTimedOut` est le code de réponse attendu pour ce scénario. La commande pour installer l'SSM Agent sur le nœud cible modifie l'ID de nœud du nœud source. Comme l'ID de nœud a changé, le nœud source n'est pas en mesure de répondre au nœud cible que la commande a échoué, s'est terminée ou a expiré lors de l'exécution.

Impossible de charger les associations de nœuds

Problème : après avoir exécuté les commandes d'installation, l'erreur suivante s'affiche dans les journaux d'erreurs de l'SSM Agent :

```
Unable to load instance associations, unable to retrieve
associations unable to retrieve associations error occurred in
```

`RequestManagedInstanceRoleToken: MachineFingerprintDoesNotMatch: Fingerprint doesn't match`

Cette erreur s'affiche lorsque l'ID de la machine ne reste pas après un redémarrage.

Solution : pour résoudre ce problème, exécutez la commande suivante. Cette commande force l'ID de la machine à rester après un redémarrage.

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

Comment installer le SSM Agent sur des Windows nœuds hybrides

Cette rubrique décrit comment installer SSM Agent sur des machines Windows Server dans un environnement hybride et multicloud. Si vous prévoyez d'utiliser des machines Linux non EC2 dans un environnement hybride et multicloud, consultez l'étape précédente, Comment installer le SSM Agent sur des nœuds Linux hybrides.

> **Important**
>
> Cette procédure concerne des machines non EC2 (Amazon Elastic Compute Cloud) dans un environnement hybride et multicloud. Pour télécharger et installer l'SSM Agent sur une instance EC2 pour Windows Server, consultez Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Windows Server.

Avant de commencer, recherchez le code d'activation et l'ID d'activation qui vous ont été envoyés à la fin de l'activation hybride, plus haut dans la rubrique Créez une activation hybride pour enregistrer les nœuds auprès de Systems Manager. Vous indiquez le code et l'ID dans la procédure suivante.

Pour installer SSM Agent sur des machines Windows Server non EC2 dans un environnement hybride et multicloud

1. Connectez-vous à un serveur ou une VM de votre environnement hybride et multicloud.
2. Si vous utilisez un proxy HTTP ou HTTPS, vous devez définir les variables d'environnement `http_proxy` ou `https_proxy` dans la session shell en cours. Si vous n'utilisez pas de proxy, vous pouvez ignorer cette étape.

   Pour un serveur proxy HTTP, définissez cette variable :

```
http_proxy=http://hostname:port
https_proxy=http://hostname:port
```

Pour un serveur proxy HTTPS, définissez cette variable :

```
http_proxy=http://hostname:port
https_proxy=https://hostname:port
```

3. Ouvrez Windows PowerShell en mode élevé (administratif).
4. Copiez et collez le bloc de commande suivant dans les Windows PowerShell. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations. Par exemple, le code d'activation et l'identifiant d'activation générés lorsque vous créez une activation hybride, et avec l'identifiant du fichier à SSM Agent partir Région AWS duquel vous souhaitez effectuer le téléchargement.

> **Note**
>
> Prenez note des informations importantes suivantes :
>
> - `ssm-setup-cli` prend en charge une option `manifest-url` qui détermine la source à partir de laquelle l'agent est téléchargé. Ne spécifiez aucune valeur pour cette option à moins que votre organisation ne l'exige.
> - Vous pouvez utiliser le script fourni ici pour valider la signature des`ssm-setup-cli`.
> - Lors de l'enregistrement des instances, n'utilisez que le lien de téléchargement fourni pour `ssm-setup-cli`. `ssm-setup-cli` ne doit pas être stocké séparément pour une utilisation ultérieure.

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

De plus, `ssm-setup-cli` inclut les options suivantes :

- `version` : les valeurs valides sont `latest` et `stable`.
- `downgrade` : rétablit une version antérieure de l'agent.

- `skip-signature-validation` : ignore la validation de signature lors du téléchargement et de l'installation de l'agent.

64-bit

```
[System.Net.ServicePointManager]::SecurityProtocol = 'TLS12'
$code = "activation-code"
$id = "activation-id"
$region = "us-east-1"
$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/ssm-setup-cli.exe", $dir + "\ssm-
setup-cli.exe")
./ssm-setup-cli.exe -register -activation-code="$code" -activation-id="$id" -
region="$region"
Get-Content ($env:ProgramData + "\Amazon\SSM\InstanceData\registration")
Get-Service -Name "AmazonSSMAgent"
```

32-bit

```
"[System.Net.ServicePointManager]::SecurityProtocol = 'TLS12'"
$code = "activation-code"
$id = "activation-id"
$region = "us-east-1"
$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/ssm-setup-cli.exe", $dir + "\ssm-setup-
cli.exe")
./ssm-setup-cli.exe -register -activation-code="$code" -activation-id="$id" -
region="$region"
Get-Content ($env:ProgramData + "\Amazon\SSM\InstanceData\registration")
Get-Service -Name "AmazonSSMAgent"
```

5. Appuyez sur `Enter`.

> **Note**
>
> Si la commande échoue, vérifiez que vous utilisez la dernière version de AWS Tools for PowerShell.

La commande exécute les opérations suivantes :

- Télécharge et installe SSM Agent sur la machine.
- Enregistre la machine avec le service Systems Manager.
- 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
```

La machine est désormais un nœud géré. Ces nœuds gérés sont à présent identifiés avec le préfixe « mi- ». Vous pouvez afficher les nœuds gérés sur la page des nœuds gérés dansFleet Manager, à l'aide de la AWS CLI commande describe-instance-informationou de la commande API DescribeInstanceInformation.

Configuration de la rotation automatique de la clé privée

Pour renforcer votre niveau de sécurité, vous pouvez configurer AWS Systems Manager Agent (SSM Agent) pour qu'il fasse automatiquement pivoter la clé privée dans un environnement hybride et multicloud. Vous pouvez accéder à cette fonction en utilisant la version 3.0.1031.0 ou ultérieure de l'SSM Agent. Procédez comme suit pour activer cette fonction.

Pour configurer SSM Agent de sorte à effectuer une rotation de la clé privée d'un environnement hybride et multicloud

1. Accédez à `/etc/amazon/ssm/` sur une machine Linux ou à `C:\Program Files\Amazon\SSM` sur une machine Windows Server.
2. Copiez le contenu de `amazon-ssm-agent.json.template` vers un nouveau fichier appelé `amazon-ssm-agent.json`. Enregistrez `amazon-ssm-agent.json` dans le même répertoire que `amazon-ssm-agent.json.template`.
3. Recherchez `Profile`, `KeyAutoRotateDays`. Saisissez le nombre de jours qui doit séparer les rotations automatiques de clé privée.
4. Redémarrez SSM Agent.

Chaque fois que vous modifiez la configuration, redémarrez l'SSM Agent.

Vous pouvez personnaliser d'autres fonctions de l'SSM Agent en suivant la même procédure. Pour une up-to-date liste des propriétés de configuration disponibles et de leurs valeurs par défaut, consultez la section Définitions des propriétés de configuration.

Désenregistrer et réenregistrer un nœud géré

Vous pouvez annuler l'enregistrement d'un nœud géré en appelant l'opération DeregisterManagedInstance API depuis Windows AWS CLI ou depuis Tools for Windows. PowerShell Voici un exemple de commande de l'interface de ligne de commande :

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

Pour supprimer les informations d'enregistrement restantes pour l'agent, supprimez la clé `IdentityConsumptionOrder` du fichier `amazon-ssm-agent.json`. Ensuite, exécutez la commande suivante :

```
amazon-ssm-agent -register -clear
```

Vous pouvez réenregistrer une machine après avoir annulé son enregistrement. Procédez comme suit pour réenregistrer une machine sous forme de nœud géré. Une fois la procédure terminée, votre nœud géré s'affiche à nouveau dans la liste des nœuds gérés.

Réenregistrer un nœud géré sur une machine hybride Windows

1. Connectez-vous à votre machine.

2. Exécutez la commande suivante. Veillez à remplacer les valeurs d'espace réservé par le code d'activation et l'ID d'activation générés lors de la création d'une activation hybride, et par l'identifiant de la région depuis laquelle vous souhaitez télécharger SSM Agent.

```
'yes' | & Start-Process ./ssm-setup-cli.exe -ArgumentList @("-register", "-
activation-code=$code", "-activation-id=$id", "-region=$region") -Wait
Get-Content ($env:ProgramData + "\Amazon\SSM\InstanceData\registration")
Get-Service -Name "AmazonSSMAgent"
```

Gestion des appareils de pointe avec Systems Manager

Cette section décrit les tâches de configuration effectuées par les administrateurs de comptes et de systèmes pour permettre la configuration et la gestion des appareils AWS IoT Greengrass principaux. Une fois ces tâches terminées, les utilisateurs auxquels l' Compte AWS administrateur a accordé des autorisations peuvent les utiliser AWS Systems Manager pour configurer et gérer les AWS IoT Greengrass principaux appareils de leur organisation.

> **Note**
>
> - SSM Agentpour AWS IoT Greengrass n'est pas pris en charge sur macOS Windows 10. Vous ne pouvez pas utiliser les fonctionnalités de Systems Manager pour gérer et configurer des appareils de périphérie utilisant ces systèmes d'exploitation.
> - Systems Manager prend également en charge les périphériques périphériques qui ne sont pas configurés en tant que périphériques AWS IoT Greengrass principaux. Pour utiliser Systems Manager afin de gérer les appareils AWS IoT principaux et les appareils non AWS périphériques, vous devez les configurer à l'aide d'une activation hybride. Pour plus d'informations, consultez Utilisation de Systems Manager dans des environnements hybrides et multicloud.
> - Pour utiliser Session Manager et la correction d'applications Microsoft avec vos appareils de périphérie, vous devez activer le niveau d'instances avancées. Pour plus d'informations, consultez Activation du niveau d'instances avancées.

Avant de commencer

Vérifiez que vos appareils de périphérie répondent aux exigences suivantes.

- Vos appareils Edge doivent répondre aux exigences pour être configurés en tant que périphériques AWS IoT Greengrass principaux. Pour plus d'informations, consultez la section Configuration des appareils AWS IoT Greengrass principaux dans le Guide du AWS IoT Greengrass Version 2 développeur.
- Vos appareils Edge doivent être compatibles avec AWS Systems Manager Agent (SSM Agent). Pour plus d'informations, consultez Systèmes d'exploitation pris en charge pour Systems Manager.
- Vos appareils de périphérie doivent être en mesure de communiquer avec le service Systems Manager dans le cloud. Systems Manager ne prend pas en charge les appareils de périphérie déconnectés.

À propos de la configuration des appareils de périphérie

La configuration AWS IoT Greengrass des appareils pour Systems Manager implique les processus suivants.

> **Note**
>
> Pour plus d'informations sur la désinstallation SSM Agent à partir d'un périphérique Edge, voir Désinstaller l' AWS Systems Manager agent dans le guide du AWS IoT Greengrass Version 2 développeur.

Créez un rôle de service IAM pour vos appareils Edge

AWS IoT Greengrass les périphériques principaux nécessitent un rôle de service AWS Identity and Access Management (IAM) avec AWS Systems Manager lequel communiquer. Le rôle accorde la `AssumeRole`confiance AWS Security Token Service (AWS STS) au service Systems Manager. Vous devez uniquement créer la fonction de service une fois pour chaque Compte AWS. Vous spécifierez ce rôle pour le `RegistrationRole` paramètre lorsque vous configurerez et déploierez le SSM Agent composant sur vos AWS IoT Greengrass appareils. Si vous avez déjà créé ce rôle lors de la configuration de nœuds non EC2 pour un environnement hybride et multicloud, vous pouvez ignorer cette étape.

> **Note**
>
> Les utilisateurs de votre entreprise ou organisation qui utilisent Systems Manager sur vos appareils de périphérie doivent recevoir l'autorisation dans IAM d'appeler l'API Systems Manager.

Exigence pour les politiques de compartiment S3

Dans les cas suivants, vous devez créer une politique d'autorisation IAM personnalisée pour les compartiments Amazon Simple Storage Service (Amazon S3) avant de terminer cette procédure :

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

  > **Note**
  >
  > Si vos appareils sont protégés par un pare-feu et que vous prévoyez d'utiliser Patch Manager, le pare-feu doit autoriser l'accès au point de terminaison du référentiel de correctifs `arn:aws:s3:::patch-baseline-snapshot-`*`région`*`/*`. *`région`* représente l'identifiant d'une région Région AWS prise en charge par AWS Systems Manager, par exemple `us-east-2` pour la région USA Est (Ohio). Pour obtenir une liste des valeurs *`region`* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la Référence générale d'Amazon Web Services.

AWS CLI

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

1. Installez et configurez le AWS Command Line Interface (AWS CLI), si ce n'est pas déjà fait.

Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l' AWS CLI.

2. Sur votre machine locale, créez un fichier texte avec un nom tel que `SSMService-Trust.json` avec la politique d'approbation suivante. Assurez-vous d'enregistrer le fichier avec l'extension de fichier `.json`.

> **Note**
>
> Notez le nom. Vous le spécifierez lorsque vous le SSM Agent déploierez sur vos appareils AWS IoT Greengrass principaux.

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

3. Ouvrez le AWS CLI, et dans le répertoire où vous avez créé le fichier JSON, exécutez la commande create-role pour créer le rôle de service. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations.

Linux et macOS

```
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
```

4. Exécutez la commande 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 vos appareils de périphérie à exécuter des commandes à l'aide de Systems Manager.

> **Note**
>
> Les politiques que vous ajoutez pour un profil de service pour des appareils de périphérie correspondent aux mêmes politiques que celles utilisées pour créer un profil d'instance pour des instances Amazon Elastic Compute Cloud (Amazon EC2). Pour plus d'informations sur les politiques IAM utilisées dans les commandes suivantes, consultez Configurer les autorisations d'instance requises pour Systems Manager.

(Obligatoire) Exécutez la commande suivante pour autoriser un périphérique périphérique à utiliser les fonctionnalités AWS Systems Manager de base du service.

Linux et macOS

```
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 politique de compartiment S3 personnalisée pour votre rôle de service, exécutez la commande suivante pour autoriser AWS Systems Manager Agent (SSM Agent) à accéder aux compartiments que vous avez spécifiés dans la politique. Remplacez *account_ID* et *my_bucket_policy_name* par l'ID de votre Compte AWS et le nom de votre compartiment.

Linux et macOS

```
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 SSM Agent à accéder à AWS Directory Service en votre nom pour les demandes de jonction de domaine par les appareils de périphérie. La fonction de service requiert uniquement cette politique si vous joignez vos appareils de périphérie à un annuaire Microsoft AD.

Linux et macOS

```
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 autoriser l' CloudWatch agent à s'exécuter sur vos appareils Edge. Cette commande permet de lire des informations sur un appareil et de les y écrire CloudWatch. Votre rôle de service n'a besoin de cette politique que si vous utilisez des services tels qu'Amazon EventBridge ou Amazon CloudWatch Logs.

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

Tools for PowerShell

Pour créer un rôle de service IAM pour un AWS IoT Greengrass environnement ()AWS Tools for Windows PowerShell

1. Installez et configurez les AWS Tools for PowerShell (Outils pour Windows PowerShell), si ce n'est pas déjà fait.

   Pour plus d'informations, consultez Installation d' AWS Tools for PowerShell.

2. Sur votre machine locale, créez un fichier texte avec un nom tel que `SSMService-Trust.json` avec la politique d'approbation suivante. Assurez-vous d'enregistrer le fichier avec l'extension de fichier `.json`.

   > **Note**
   >
   > Notez le nom. Vous le spécifierez lorsque vous le SSM Agent déploierez sur vos appareils AWS IoT Greengrass principaux.

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

3. Ouvrez PowerShell en mode administratif, et dans le répertoire où vous avez créé le fichier JSON, exécutez New-IAMRole comme suit pour créer un rôle de service.

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

4. Utilisez Register-IAM RolePolicy comme suit pour autoriser le rôle de service que vous avez créé à créer un jeton de session. Ce jeton de session autorise vos appareils de périphérie à exécuter des commandes à l'aide de Systems Manager.

> **Note**
>
> Les politiques que vous ajoutez pour une fonction de service pour des appareils de périphérie dans un environnement AWS IoT Greengrass sont les mêmes politiques que celles utilisées pour créer un profil d'instance pour des instances EC2. Pour plus d'informations sur les AWS politiques utilisées dans les commandes suivantes, consultez Configurer les autorisations d'instance requises pour Systems Manager.

(Obligatoire) Exécutez la commande suivante pour autoriser un périphérique périphérique à utiliser les fonctionnalités AWS Systems Manager de base du service.

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

Si vous avez créé une politique de compartiment S3 personnalisée pour votre rôle de service, exécutez la commande suivante pour permettre à SSM Agent d'accéder aux compartiments que vous avez spécifiés dans la politique. Remplacez *account_ID* et *my_bucket_policy_name* par l'ID de votre Compte AWS 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 SSM Agent à accéder à AWS Directory Service en votre nom pour les demandes de jonction de domaine par les appareils de périphérie. La fonction de service requiert uniquement cette politique si vous joignez vos appareils de périphérie à un annuaire Microsoft AD.

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

(Facultatif) Exécutez la commande suivante pour autoriser l' CloudWatch agent à s'exécuter sur vos appareils Edge. Cette commande permet de lire des informations sur un appareil et

de les y écrire CloudWatch. Votre rôle de service n'a besoin de cette politique que si vous utilisez des services tels qu'Amazon EventBridge ou Amazon CloudWatch Logs.

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

Configurez vos appareils Edge pour AWS IoT Greengrass

Configurez vos appareils Edge en tant qu'appareils AWS IoT Greengrass principaux. Le processus de configuration implique la vérification des systèmes d'exploitation pris en charge et de la configuration requise, ainsi que l'installation et la configuration du logiciel AWS IoT Greengrass Core sur vos appareils. Pour plus d'informations, consultez Configuration des appareils Core AWS IoT Greengrass dans le Guide du développeur AWS IoT Greengrass Version 2 .

Mettez à jour le rôle d'échange de AWS IoT Greengrass jetons et installez-le SSM Agent sur vos appareils Edge

La dernière étape de l'installation et de la configuration de vos appareils AWS IoT Greengrass principaux pour Systems Manager nécessite que vous mettiez à jour le rôle de service des appareils AWS IoT Greengrass AWS Identity and Access Management (IAM), également appelé rôle d'échange de jetons, et que vous déployiez AWS Systems Manager l'agent (SSM Agent) sur vos AWS IoT Greengrass appareils. Pour plus d'informations sur ces processus, consultez Installation de l'agent AWS Systems Manager dans le Guide du développeur AWS IoT Greengrass Version 2 .

Après le déploiement SSM Agent sur vos appareils, les enregistre AWS IoT Greengrass automatiquement auprès de Systems Manager. Aucun enregistrement supplémentaire n'est nécessaire. Vous pouvez commencer à utiliser les fonctionnalités de Systems Manager pour accéder à vos AWS IoT Greengrass appareils, les gérer et les configurer.

> **Note**
>
> Vos appareils de périphérie doivent être en mesure de communiquer avec le service Systems Manager dans le cloud. Systems Manager ne prend pas en charge les appareils de périphérie déconnectés.

Création d'un administrateur AWS Organizations délégué pour Systems Manager

Lorsque vous configurez une organisation dans AWS Organizations, vous attribuez un compte de gestion pour effectuer toutes les tâches administratives pour tous Services AWS. L'utilisateur du compte de gestion peut attribuer un compte d'administrateur délégué uniquement à Systems Manager pour effectuer des tâches administratives pour Change ManagerExplorer, etOpsCenter. AWS Organizations est un service de gestion de comptes que vous pouvez utiliser pour créer une organisation et Comptes AWS attribuer la gestion centralisée de ces comptes. Pour plus d'informations à ce sujet AWS Organizations, consultez AWS Organizationsle guide de AWS Organizations l'utilisateur.

Change ManagerExplorer, et les OpsCenter fonctionnalités de AWS Systems Manager, travaillez avec AWS Organizations pour effectuer des tâches sur tous les comptes membres de votre organisation. Vous ne pouvez désigner qu'un seul administrateur délégué pour toutes les fonctionnalités de Systems Manager. Le compte d'administrateur délégué doit être le membre de l'unité d'organisation à laquelle il est affecté.

Rubriques

Utiliser un administrateur délégué avec Change Manager

Change Manager est un cadre de gestion des modifications d'entreprise pour demander, approuver, implémenter et signaler des modifications opérationnelles apportées à la configuration et à l'infrastructure de votre application.

Si vous utilisez Change Manager au sein d'une organisation, désignez un compte d'administrateur délégué pour gérer les modèles, approbations et rapports de modifications pour tous les comptes membres. À l'aide de la configuration rapide, vous pouvez configurer l'utilisation de Change Manager avec une organisation et sélectionner le compte d'administrateur délégué. Si vous l'utilisez Change Manager avec un seul Compte AWS, le compte d'administrateur délégué n'est pas requis.

Par défaut, Change Manager affiche toutes les tâches liées aux modifications dans le compte d'administrateur délégué. Pour obtenir des instructions sur la configuration d'un administrateur

délégué lors de la mise en place de Change Manager pour une organisation, veuillez consulter la rubrique Configuration de Change Manager pour une organisation (compte de gestion).

> **Important**
>
> Si vous utilisez Change Manager au sein d'une organisation, nous vous recommandons de toujours apporter les modifications à partir du compte d'administrateur délégué. Bien qu'il soit possible d'apporter des modifications à partir d'autres comptes de l'organisation, celles-ci ne seront pas signalées ou affichées à partir du compte d'administrateur délégué.

Utiliser un administrateur délégué avec Explorer

Explorerest un tableau de bord des opérations personnalisable qui fournit une vue agrégée des données d'exploitation (OpsData) pour votre Comptes AWS, à travers Régions AWS.

Vous pouvez configurer un compte d'administrateur délégué pour que Systems Manager puisse agréger Explorer les données provenant de plusieurs régions et comptes en utilisant la synchronisation des données de ressources avec AWS Organizations. Un administrateur délégué peut rechercher, filtrer et agréger Explorer des données à l'aide du AWS Management Console, du AWS Command Line Interface (AWS CLI) ou AWS Tools for Windows PowerShell.

Lorsque vous utilisez un compte d'administrateur délégué pour Explorer, vous limitez le nombre d'administrateurs qui peuvent créer ou supprimer des synchronisations de données de ressources de plusieurs comptes et régions à un seul Compte AWS.

Vous pouvez synchroniser les données d'exploitation Comptes AWS dans l'ensemble de votre organisation en utilisantExplorer. Pour découvrir comment désigner un administrateur délégué à partir d'Explorer, veuillez consulter la rubrique Configuration d'un administrateur délégué.

Utiliser un administrateur délégué avec OpsCenter

OpsCenterfournit un emplacement central où les ingénieurs des opérations et les professionnels de l'informatique peuvent gérer les éléments de travail opérationnels (OpsItems) liés aux AWS ressources. Si vous souhaitez utiliser OpsCenter pour gérer OpsItems de manière centralisée entre les comptes, vous devez configurer l'organisation dans AWS Organizations.

En utilisant Quick Setup pour OpsCenter, vous pouvez attribuer un compte administrateur délégué et configurer OpsCenter pour gérer OpsItems de manière centralisée. Pour plus d'informations,

voir (Facultatif) Configurer OpsCenter pour gérer OpsItems sur plusieurs comptes à l'aide de Quick Setup.

Configuration générale pour AWS Systems Manager

Si ce n'est pas déjà fait, inscrivez-vous Compte AWS et créez un utilisateur administratif.

Inscrivez-vous pour un Compte AWS

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

Pour vous inscrire à un Compte AWS

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

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

   Lorsque vous vous inscrivez à un Compte AWS, un Utilisateur racine d'un compte AWSest créé. Par défaut, seul l'utilisateur racine a accès à l'ensemble des Services AWS et des ressources de ce compte. Pour des raisons de sécurité, attribuez un accès administratif à un utilisateur et utilisez uniquement l'utilisateur root pour effectuer les tâches nécessitant un accès utilisateur root.

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

Création d'un utilisateur doté d'un accès administratif

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

Sécurisez votre Utilisateur racine d'un compte AWS

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

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

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

Création d'un utilisateur doté d'un accès administratif

1. Activez IAM Identity Center.

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

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

Connectez-vous en tant qu'utilisateur disposant d'un accès administratif

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

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

Attribuer l'accès à des utilisateurs supplémentaires

1. Dans IAM Identity Center, créez un ensemble d'autorisations conforme aux meilleures pratiques en matière d'application des autorisations du moindre privilège.

Pour obtenir des instructions, voir Création d'un ensemble d'autorisations dans le guide de AWS IAM Identity Center l'utilisateur.

2. Affectez des utilisateurs à un groupe, puis attribuez un accès d'authentification unique au groupe.

   Pour obtenir des instructions, voir Ajouter des groupes dans le guide de AWS IAM Identity Center l'utilisateur.

Réaliser une tâche de gestion avec Systems Manager

Utilisez ce didacticiel pour commencer AWS Systems Manager. Vous allez apprendre à lancer une instance Amazon Elastic Compute Cloud (Amazon EC2) qui est gérée par Systems Manager, et à plus à vous connecter à l'instance gérée.

Systems Manager étant un ensemble de plusieurs fonctionnalités, aucun didacticiel ou procédure ne peut présenter l'ensemble du service. Ce didacticiel fournit une introduction à certaines fonctionnalités.

Prérequis

Avant de commencer, assurez-vous d'avoir terminé les étapes de Utilisation de Systems Manager avec des instances EC2.

Lancez une instance à l'aide d'une AMI avec un SSM Agent préinstallé

Vous pouvez lancer une instance Amazon EC2 à l'aide de la procédure AWS Management Console décrite dans la procédure suivante. Ce didacticiel a pour but de vous aider à lancer rapidement votre première instance gérée. Il ne couvrira donc pas toutes les options possibles.

Pour lancer une instance

1. Ouvrez la console Amazon EC2 à l'adresse https://console.aws.amazon.com/ec2/.
2. Sur le tableau de bord de la console EC2, dans la case Launch instance (Lancer une instance), choisissez Launch instance (Lancer une instance), puis Launch instance (Lancer une instance) dans les options qui s'affichent.
3. Sous Noms et balises, pour Nom, saisissez un nom descriptif pour votre instance.
4. Sous Images d'applications et de systèmes d'exploitation (Amazon Machine Image), procédez comme suit :

    a. Choisissez Démarrage rapide, puis Amazon Linux. Il s'agit du système d'exploitation (OS) de votre instance.

    b. Pour Amazon Machine Image (AMI), choisissez une version Amazon Linux 2 (HVM).

5. Sous Type d'instance, dans la liste Type d'instance, vous pouvez sélectionner la configuration matérielle de votre instance. Sélectionnez le type d'instance `t2.micro` sélectionné par défaut. Le type d'`t2.micro`instance est éligible au niveau AWS gratuit. Dans les Régions AWS où `t2.micro` n'est pas disponible, vous pouvez utiliser une instance `t3.micro` avec l'offre gratuite. Pour plus d'informations, consultez la page sur l'offre gratuite AWS.
6. Sous Paire de clés (connexion), pour Nom de la paire de clés, choisissez votre paire de clés.
7. Sous Paramètres réseau, choisissez Modifier. Pour Nom du groupe de sécurité, vous pouvez remarquer que l'assistant a créé et sélectionné un groupe de sécurité pour vous. Vous pouvez utiliser ce groupe de sécurité ou sélectionner un groupe de sécurité que vous avez créé précédemment en procédant comme suit :
   a. Choisissez Select existing security group (Sélectionner un groupe de sécurité existant).
   b. Depuis Common security groups (Groupes de sécurité communs), choisissez votre groupe de sécurité dans la liste des groupes de sécurité existants.
8. Si vous n'utilisez pas la Configuration de gestion des hôtes par défaut, développez la section Détails avancés et, sous Profil d'instance IAM, choisissez le profil d'instance que vous avez créé lors de la configuration dans Configurer les autorisations d'instance requises pour Systems Manager.
9. Conservez les sélections par défaut des autres paramètres de configuration de votre instance.
10. Consultez le résumé de la configuration de votre instance dans le volet Récapitulatif. Une fois que vous êtes prêt, choisissez Lancer l'instance.
11. Une page de confirmation vous indique que votre instance est en cours de lancement. Sélectionnez View all instances (Afficher toutes les instances) pour fermer la page de confirmation et revenir à la console.
12. Sur l'écran Instances, vous pouvez afficher le statut du lancement. Il suffit de peu de temps pour lancer une instance.
13. Vous devrez peut-être patienter quelques minutes avant que l'instance n'apparaisse comme gérée et que vous ne puissiez vous y connecter. Vérifiez que votre instance a réussi ses contrôles de statut ; vous pouvez voir cette information dans la colonne Contrôle de statut.

Connectez-vous à votre instance gérée à l'aide de Systems Manager

Pour vous connecter à votre instance gérée, procédez comme suit :

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Cliquez sur le bouton situé à côté de l'instance à laquelle vous souhaitez vous connecter.
4. Dans le menu Actions du nœud, choisissez Démarrer une session de terminal.
5. Cliquez sur Connect (Connexion).

Nettoyez votre instance

Si vous en avez fini avec l'instance gérée que vous avez créée pour ce didacticiel, résiliez-la. Résilier une instance la supprime définitivement.

Pour mettre fin à une instance

1. Ouvrez la console Amazon EC2 à l'adresse https://console.aws.amazon.com/ec2/.
2. Dans le panneau de navigation, sélectionnez Instances. Sélectionnez l'instance dans la liste des instances.
3. Choisissez État de l'instance, Résilier l'instance.
4. Choisissez Résilier lorsque vous êtes invité à confirmer.

   Amazon EC2 arrête et met fin à votre instance. Une fois votre instance résiliée, elle reste visible sur la console pendant un court instant, puis l'entrée est supprimée automatiquement. Vous ne pouvez pas supprimer vous-même l'instance résiliée de l'affichage de la console.

Utilisation de l'option SSM Agent

AWS Systems Manager Agent (SSM Agent) est un logiciel Amazon qui s'exécute sur des instances Amazon Elastic Compute Cloud (Amazon EC2), des appareils périphériques, des serveurs sur site et des machines virtuelles (VM). SSM Agentpermet 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 AWS Cloud, puis les exécute comme indiqué dans la demande. SSM Agentrenvoie ensuite les informations d'état et d'exécution au service Systems Manager en utilisant le Amazon Message Gateway Service(`ssmmessages`). (En cas de Régions AWS lancement avant 2024, les informations d'état et d'exécution peuvent également être renvoyées par le Amazon Message Delivery Service(préfixe de service :`ec2messages`).)

Si vous surveillez le trafic, vous constaterez que vos nœuds gérés communiquent avec les points de `ssmmessages.*` terminaison et éventuellement avec les points de `ec2messages.*` terminaison. Pour plus d'informations, consultez Référence : ec2messages, ssmmessages et autres opérations d'API. Pour plus d'informations sur le portage SSM Agent des journaux vers Amazon CloudWatch Logs, consultezSurveillance AWS Systems Manager.

Table des matières

Découvrez les détails techniques du SSM Agent

Utilisez les informations de cette rubrique pour vous aider à implémenter AWS Systems Manager Agent (SSM Agent) et à comprendre le fonctionnement de l'agent.

Rubriques

- SSM Agentmises à jour continues par Régions AWS
- Communications de l'SSM Agent avec des compartiments S3 gérés par AWS
- Rechercher AMIs avec le SSM Agent préinstallé
- Utilisation de SSM Agent sur des instances EC2 pour Linux
- Utilisation de SSM Agent sur des instances EC2 pour macOS
- Utilisation de SSM Agent sur des instances EC2 pour Windows Server
- Vérification du statut de l'SSM Agent et démarrage de l'agent
- Vérification du numéro de version de l'SSM Agent
- Affichage des journaux SSM Agent
- Limitation de l'accès aux commandes de niveau racine via l'SSM Agent
- Automatisation des mises à jour de l'SSM Agent
- Abonnement aux notifications SSM Agent

Comportement des informations d'identification SSM Agent version 3.2.x.x

SSM Agent stocke un ensemble d'informations d'identification temporaires dans `/var/lib/amazon/ssm/credentials` (pour Linux et macOS) ou `%PROGRAMFILES%\Amazon\SSM\credentials` (pour Windows Server) lorsqu'une instance est intégrée à l'aide de la configuration de gestion d'hôte par défaut dans Quick Setup. Les informations d'identification temporaires disposent des autorisations que vous avez spécifiées pour le rôle IAM que vous avez choisi pour la configuration de la gestion de l'hôte par défaut. Sous Linux, seul le compte root peut accéder à ces informations d'identification. Sous Windows Server, seuls le compte SYSTEM et les Administrateurs locaux peuvent accéder à ces informations d'identification.

SSM AgentPriorité des informations d'identification de l'

Cette rubrique décrit des informations importantes sur la façon dont l'SSM Agent est autorisé à exécuter des actions sur vos ressources.

> **Note**
>
> La prise en charge des appareils de périphérie est légèrement différente. Vous devez configurer vos appareils Edge pour utiliser le logiciel AWS IoT Greengrass Core, configurer un rôle de service AWS Identity and Access Management (IAM) et déployer sur vos appareils

SSM Agent à l'aide AWS IoT Greengrass de. Pour plus d'informations, consultez Gestion des appareils de pointe avec Systems Manager.

Quand SSM Agent est installé sur une machine, il a besoin d'autorisations pour communiquer avec le service Systems Manager. Sur les instances Amazon Elastic Compute Cloud (Amazon EC2), ces autorisations sont fournies dans un profil d'instance attaché à l'instance. Sur une machine non EC2, SSM Agent obtient normalement les autorisations nécessaires à partir du fichier d'informations d'identification partagées qui se trouve dans `/root/.aws/credentials` (Linux et macOS) ou `%USERPROFILE%\.aws\credentials` (Windows Server). Les autorisations nécessaires sont ajoutées à ce fichier durant le processus d'activation hybride.

Dans de rares cas, cependant, des autorisations peuvent être ajoutées à une machine, à un plus grand nombre d'emplacements que ceux où SSM Agent vérifie les autorisations pour exécuter ses tâches.

Par exemple, imaginons que vous ayez configuré une instance EC2 de sorte qu'elle soit gérée par Systems Manager. Cette configuration inclut l'attachement d'un profil d'instance. Mais vous décidez ensuite d'utiliser cette instance pour les tâches de développeur ou d'utilisateur final, et d'installer l' AWS Command Line Interface (AWS CLI) par-dessus. Dans ce cas, des autorisations supplémentaires sont ajoutées à un fichier d'informations d'identification sur l'instance.

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

> **Note**
>
> Sous Linux et macOS, l'SSM Agent s'exécute en tant qu'utilisateur racine. Par conséquent, les variables d'environnement et le fichier d'informations d'identification que SSM Agent recherche dans ce processus sont ceux de l'utilisateur root uniquement (`/root/.aws/credentials`). SSM Agent n'examine pas les variables d'environnement ou le fichier d'informations d'identification d'autres utilisateurs sur l'instance lorsqu'il recherche des informations d'identification.

La chaîne de fournisseur par défaut recherche des informations d'identification dans cet ordre :

1. Variables d'environnement, si elles sont configurées (`AWS_ACCESS_KEY_ID` et `AWS_SECRET_ACCESS_KEY`).
2. Fichier d'informations d'identification partagées (`$HOME/.aws/credentials` pour Linux et macOS ou `%USERPROFILE%\.aws\credentials`pour Windows Server) avec les autorisations fournies, par exemple, par une activation hybride ou une installation de la AWS CLI .
3. Rôle AWS Identity and Access Management (IAM) pour les tâches en présence d'une application utilisant une définition RunTask de tâche ou une opération d'API Amazon Elastic Container Service (Amazon ECS).
4. Un profil d'instance IAM attaché à une instance Amazon EC2
5. Le rôle IAM choisi pour la configuration de la gestion de l'hôte par défaut.

Pour plus d'informations, consultez les rubriques connexes suivantes :

- Profils d'instance pour les instances EC2 : configurez les autorisations d'instance requises pour Systems Manager
- Activations hybrides — Créez une activation hybride pour enregistrer les nœuds auprès de Systems Manager
- AWS CLI informations d'identification — Configuration et paramètres des fichiers d'identification dans le guide de l'AWS Command Line Interface utilisateur
- Chaîne de fournisseur d'informations d'identification par défaut : Spécification d'informations d'identification dans le Manuel du développeur AWS SDK for Go

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

À propos du compte local ssm-user

À partir de la version 2.3.50.0 de l'SSM Agent, l'agent crée un compte utilisateur local appelé `ssm-user` et l'ajoute au répertoire `/etc/sudoers.d` (Linux et macOS) ou au groupe Administrateurs (Windows Server). Sur les versions de l'agent antérieures à 2.3.612.0, le compte est créé la première fois que l'SSM Agent 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 par défaut lorsqu'une session démarreSession Manager, une fonctionnalité de AWS Systems Manager. 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'SSM Agent est désinstallé.

Sur Windows Server, l'SSM Agent 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 SSM Agent 2.3.612.0, 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, créez le compte `ssm-user` manuellement, s'il n'existe pas déjà, et affectez les autorisations d'administrateur de domaine à l'utilisateur.

> **Important**
>
> Pour que le compte `ssm-user` soit créé, le profil d'instance attaché à l'instance doit fournir les autorisations requises. Pour plus d'informations, consultez Étape 2 : vérifier ou ajouter des autorisations d'instance pour Session Manager.

SSM Agent et le Instance Metadata Service (IMDS)

Systems Manager s'appuie sur les métadonnées d'instance EC2 pour fonctionner correctement. Systems Manager peut accéder aux métadonnées d'instance en utilisant la version 1 ou la version 2 d'Instance Metadata Service (IMDSv1 et IMDSv2). Votre instance doit pouvoir accéder à l'adresse IPv4 du service des métadonnées d'instance : 169.254.169.254. Pour plus d'informations, consultez Métadonnées d'instance et données utilisateur dans le Guide de l'utilisateur Amazon EC2.

Garder SSM Agent up-to-date

Une nouvelle version de SSM Agent est lancée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Le fait de ne pas utiliser la dernière version de l'agent peut empêcher votre nœud géré d'utiliser diverses capacités et fonctionnalités de Systems Manager. C'est pourquoi nous vous recommandons

d'automatiser le processus permettant de maintenir SSM Agent à jour sur vos machines. Pour plus d'informations, consultez Automatisation des mises à jour de l'SSM Agent. Abonnez-vous à la page des notes de SSM Agent publication GitHub pour recevoir des notifications concernant les SSM Agent mises à jour.

> **Note**
>
> Une nouvelle version de SSM Agent est lancée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Le fait de ne pas utiliser la dernière version de l'agent peut empêcher votre nœud géré d'utiliser diverses capacités et fonctionnalités de Systems Manager. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir SSM Agent à jour sur vos machines. Pour plus d'informations, consultez Automatisation des mises à jour de l'SSM Agent. Abonnez-vous à la page des notes de SSM Agent publication GitHub pour recevoir des notifications concernant les SSM Agent mises à jour.
> Les Amazon Machine Images (AMIs) qui comprennent l'SSM Agent par défaut peuvent prendre jusqu'à deux semaines pour publier l'AMI mis à jour avec la dernière version de l'SSM Agent. Nous vous recommandons de configurer encore plus fréquemment des mises à jour automatiques de l'SSM Agent.

S'assurer que le répertoire d'installation SSM Agent ne soit pas modifié, déplacé ou supprimé

SSM Agent est installé sur `/var/lib/amazon/ssm/` (Linux et macOS) et `%PROGRAMFILES%\Amazon\SSM\` (Windows Server). Ces répertoires d'installation contiennent des fichiers et des dossiers critiques utilisés par SSM Agent, tels qu'un fichier d'informations d'identification, des ressources pour la communication interprocessus (IPC) et des dossiers d'orchestration. Aucun élément du répertoire d'installation ne doit être modifié, déplacé ou supprimé. Dans le cas contraire, SSM Agent pourrait cesser de fonctionner correctement.

SSM Agentmises à jour continues par Régions AWS

Une fois qu'une SSM Agent mise à jour est disponible dans son GitHub référentiel, cela peut prendre jusqu'à deux semaines avant que la version mise à jour ne soit déployée auprès de tous Régions AWS à des moments différents. Pour cette raison, vous pouvez recevoir le message d'erreur « Non pris en charge sur la plate-forme actuelle » ou « Mise amazon-ssm-agent à jour vers une ancienne

version, veuillez activer l'autorisation de rétrogradation » lorsque vous tentez de déployer une nouvelle version de SSM Agent dans une région.

Pour déterminer votre version disponible de SSM Agent, vous pouvez exécuter une commande `curl`.

Pour afficher la version de l'agent disponible dans le compartiment de téléchargement global, exécutez la commande suivante.

```
curl https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/VERSION
```

Pour afficher la version de l'agent disponible dans une région particulière, exécutez la commande suivante, en remplaçant la *`région`* par celle où vous travaillez, par exemple, `us-east-2` pour la région USA Est (Ohio).

```
curl https://s3.region.amazonaws.com/amazon-ssm-region/latest/VERSION
```

Vous pouvez aussi ouvrir le fichier `VERSION` directement dans votre navigateur sans exécuter de commande `curl`.

Communications de l'SSM Agent avec des compartiments S3 gérés par AWS

Au cours de l'exécution de diverses opérations de Systems Manager, AWS Systems Manager Agent (SSM Agent) accède à un certain nombre de compartiments Amazon Simple Storage Service (Amazon S3). Ces compartiments S3 sont accessibles au public et, par défaut, l'SSM Agent s'y connecte en utilisant des appels `HTTP`.

Toutefois, si vous utilisez un point de terminaison de cloud privé virtuel (VPC) dans le cadre de vos opérations Systems Manager, vous devez fournir une autorisation explicite dans un profil d'instance Amazon Elastic Compute Cloud (Amazon EC2) pour Systems Manager, ou dans un rôle de service pour les machines non EC2 dans un environnement hybride et multicloud. Sinon, vos ressources ne peuvent pas accéder à ces compartiments publics.

Pour accorder à vos nœuds gérés l'accès à ces compartiments lorsque vous utilisez un point de terminaison d'un VPC, vous créez une politique d'autorisations Amazon S3 personnalisée, puis l'attachez à votre profil d'instance (pour les instances EC2) ou à votre fonction du service (pour les nœuds gérés non EC2).

Pour plus d'informations sur l'utilisation d'un point de terminaison de cloud privé virtuel (VPC) dans vos opérations de Systems Manager, consultez Améliorer la sécurité des instances EC2 en utilisant des points de terminaison VPC pour Systems Manager.

> **Note**
>
> Ces autorisations fournissent uniquement l'accès aux compartiments AWS gérés requis parSSM Agent. 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 plus d'informations, consultez les rubriques suivantes :

- Configurer les autorisations d'instance requises pour Systems Manager
- Créez le rôle de service IAM requis pour Systems Manager dans les environnements hybrides et multicloud

Table des matières

Autorisations de compartiment nécessaires

Le tableau suivant décrit chacun des compartiments S3 auxquels l'SSM Agent peut avoir besoin d'accéder pour des opérations Systems Manager.

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

Autorisations Amazon S3 requises par l'SSM Agent

ARN de compartiment S3	Description
`arn:aws:s3:::aws-windows-downloads-` *region*`/*`	Obligatoire pour certains documents SSM qui ne prennent en charge que les systèmes d'exploitation Windows Server, ainsi que pour d'autres pour la prise en charge multiplateforme, tels que `AWSEC2-ConfigureSTIG` .
`arn:aws:s3:::amazon-ssm-` *region*`/*`	Requise pour la mise à jour des installations de l'SSM Agent. Ces compartiments contiennent les packages d'installation de l'SSM Agent, 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'SSM Agent effectue un appel HTTP pour télécharger la mise à jour.
`arn:aws:s3:::amazon-ssm-packages-` *region*`/*`	Requise pour utiliser les versions de l'SSM Agent antérieures à la version 2.2.45.0 en vue d'exécuter le document SSM `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'SSM Agent. Ce service est utilisé pour exécuter le document `AWS-ConfigureAWSPackage` . Cette autorisation est nécessaire pour tous Régions AWS sauf pour la région Afrique (Le Cap) (af-south-1) et pour la région Europe (Milan) (eu-south-1).
`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'SSM Agent. Ce service est

ARN de compartiment S3	Description
	utilisé pour exécuter le document SSM `AWS-ConfigureAWSPackage` . Cette autorisation est nécessaire uniquement pour la région Afrique (Le Cap) (af-sud-1) et la région Europe (Milan) (eu-sud 1).
`arn:aws:s3:::aws-ssm-document-attachments-` *region*`/*`	Fournit un accès au compartiment S3 contenant les packages pourDistributor, une fonctionnalité de AWS Systems Manager, détenus par AWS.

ARN de compartiment S3	Description
`arn:aws:s3:::patch-baseline-snapshot-` *region*`/*`	Fournit l'accès au compartiment S3 contenant les instantanés de référentiel de correctifs. Ceci est obligatoire si vous utilisez l'un des documents SSM suivants : • `AWS-RunPatchBaseline` • `AWS-RunPatchBaselineAssociation` • `AWS-RunPatchBaselineWithHooks` • `AWS-ApplyPatchBaseline` (un document SSM hérité) **Note** Dans la région Moyen-Orient (Bahreïn) (me-south-1) uniquement, ce compartiment S3 utilise une convention de dénomination différente. Pour cette Région AWS uniquement, utilisez plutôt le compartiment suivant. • `patch-baseline-snapshot-me-south-1-uduvl7q8` Dans la région Afrique (Le Cap) (me-south-1) uniquement, ce compartiment S3 utilise une convention de dénominat ion différente. Pour cette Région AWS uniquement, utilisez plutôt le compartim ent suivant. • `patch-baseline-snapshot-af-south-1-tbxdb5b9`

ARN de compartiment S3	Description
Pour les nœuds gérés Linux et Windows Server : `arn:aws:s3:::aws-ssm-` *region*`/*` Pour les instances Amazon EC2macOS : `arn:aws:s3:::aws-patchmanager-macos-` *region*`/*`	Fournit l'accès au compartiment S3 contenant les modules requis à utiliser avec certains documents Systems Manager (documents SSM). Par exemple : • `arn:aws:s3:::aws-ssm-us-east-2/*` • `aws-patchmanager-macos-us-east-2/*` Exceptions Dans certains cas, les noms des compartiments S3 Régions AWS utilisent une convention de dénomination étendue, comme le montrent leurs ARN. Pour ces régions, utilisez plutôt les ARN suivants : • Middle East (Bahrain) Region (me-south-1)) : `aws-patch-manager-me-south-1-a53fc9dce` • Africa (Cape Town) Region (af-south-1) : `aws-patch-manager-af-south-1-bdd5f65a9` • Europe (Milan) Region (eu-south-1) : `aws-patch-manager-eu-south-1-c52f3f594` • Asia Pacific (Osaka) Region (ap-northeast-3) : `aws-patch-manager-ap-northeast-3-67373598a` Documents SSM

ARN de compartiment S3	Description
	Voici quelques documents SSM couramment utilisés, stockés dans ces compartiments. Dans `arn:aws:s3:::aws-ssm-` *region*`/`: • `AWS-RunPatchBaseline` • `AWS-RunPatchBaselineAssociation` • `AWS-RunPatchBaselineWithHooks` • `AWS-InstanceRebootWithHooks` • `AWS-ConfigureWindowsUpdate` • `AWS-FindWindowsUpdates` • `AWS-PatchAsgInstance` • `AWS-PatchInstanceWithRollback` • `AWS-UpdateSSMAgent` • `AWS-UpdateEC2Config` Dans `arn:aws:s3:::aws-patchmanager-macos-` *region*`/`: • `AWS-RunPatchBaseline` • `AWS-RunPatchBaselineAssociation` • `AWS-RunPatchBaselineWithHooks` • `AWS-InstanceRebootWithHooks` • `AWS-PatchAsgInstance` • `AWS-PatchInstanceWithRollback`

Exemple

L'exemple suivant illustre comment fournir l'accès aux compartiments S3 requis pour les opérations Systems Manager dans la région USA Est (Ohio) (us-east-2). Dans la plupart des cas, vous devez

fournir ces autorisations explicitement dans un profil d'instance ou un rôle de service uniquement lors de l'utilisation d'un point de terminaison de VPC.

> **Important**
>
> Dans cette politique, 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 l'accès aux compartiments S3 vers lesquels vous ne prévoyez pas 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.

```
{
        "Version": "2012-10-17",
        "Statement": [
            {
                "Effect": "Allow",
                "Action": "s3:GetObject",
                "Resource": [
                    "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:::aws-ssm-document-attachments-us-east-2/*",
                    "arn:aws:s3:::patch-baseline-snapshot-us-east-2/*",
                    "arn:aws:s3:::aws-ssm-us-east-2/*",
                    "arn:aws:s3:::aws-patchmanager-macos-us-east-2/*"
                ]
            }
        ]
    }
```

Validation des machines activées par un système hybride à l'aide d'une empreinte matérielle

Lorsque vous exécutez des machines non EC2 dans un environnement hybride et multicloud, SSM Agent rassemble des attributs système (autrement appelés hachage matériel), qu'il utilise pour calculer une empreinte digitale. L'empreinte digitale est une chaîne opaque que l'agent transmet à certaines API Systems Manager. Cette empreinte digitale unique associe l'appelant à un nœud géré

et activé par un système hybride particulier. L'agent stocke l'empreinte digitale et le hachage matériel sur le disque local, à un emplacement désigné comme Coffre-fort.

L'agent calcule le hachage matériel et l'empreinte digitale lorsque la machine est enregistrée pour une utilisation avec Systems Manager. Ensuite, l'empreinte digitale est transmise au service Systems Manager lorsque l'agent envoie une commande `RegisterManagedInstance`.

Plus tard, lors de l'envoi d'une commande `RequestManagedInstanceRoleToken`, l'agent vérifie l'empreinte digitale et le hachage matériel dans le coffre-fort afin de s'assurer que les attributs de la machine actuelle correspondent au hachage matériel stocké. Si les attributs de la machine actuelle correspondent au hachage matériel stocké dans le coffre-fort, l'agent transmet l'empreinte digitale du coffre-fort à une `RegisterManagedInstance`, et l'appel est alors considéré comme réussi.

Si les attributs de la machine actuelle ne correspondent pas au hachage matériel stocké, l'SSM Agent calcule une nouvelle empreinte digitale, stocke le nouveau hachage matériel et l'empreinte digitale dans le coffre-fort et transmet la nouvelle empreinte digitale à un `RequestManagedInstanceRoleToken`. Cela provoque l'échec du *`RequestManagedInstanceRoleToken`*, et l'agent ne pourra pas obtenir un jeton de rôle pour se connecter au service Systems Manager.

Cet échec est intégré, et il sert d'étape de vérification pour empêcher que plusieurs nœuds gérés communiquent avec le service Systems Manager en tant que même nœud géré.

Lorsqu'il compare les attributs de la machine actuelle au hachage matériel stocké dans le coffre-fort, l'agent utilise la logique suivante pour déterminer si l'ancien et le nouveau hachages correspondent :

- Si le SID (ID système/machine) est différent, alors ils ne correspondent pas.
- Si l'adresse IP est la même, alors ils correspondent.
- Sinon, le pourcentage d'attributs de la machine qui correspondent est calculé et comparé au seuil de similarité configuré par l'utilisateur pour déterminer s'il y a correspondance.

Le seuil de similarité est stocké dans le coffre-fort, et fait partie du hachage matériel.

Le seuil de similarité peut être défini après qu'une instance a été enregistrée en utilisant une commande semblable à celle qui suit.

Sur les machines Linux :

```
sudo amazon-ssm-agent -fingerprint -similarityThreshold 1
```

Sur Windows Server les machines utilisant PowerShell :

```
cd "C:\Program Files\Amazon\SSM\" `
    .\amazon-ssm-agent.exe -fingerprint -similarityThreshold 1
```

> **Important**
>
> Si l'un des composants utilisés pour calculer l'empreinte digitale change, cela peut entraîner la mise en veille prolongée de l'agent. Pour éviter cette mise en veille prolongée, définissez le seuil de similitude à une valeur faible, **1** par exemple.

SSM Agent sur GitHub

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

Rechercher AMIs avec le SSM Agent préinstallé

AWS Systems Manager L'agent (SSM Agent) est préinstallé sur certains Amazon Machine Images (AMIs) fournis par AWS des tiers de confiance.

Par exemple, lorsque vous lancez une instance Amazon Elastic Compute Cloud (Amazon EC2) créée à partir d'une AMI avec l'un des systèmes d'exploitation suivants, vous constaterez probablement que l'SSM Agent est déjà installé :

- AlmaLinux
- AMI de base Amazon Linux 1 datées du 09/2017 et versions ultérieures
- Amazon Linux 2
- AMIs Amazon Linux 2 Base optimisées ECS
- Amazon Linux 2023 (AL2023)
- Amazon EKS optimisé Amazon Linux AMIs
- macOS10,14.x (Mojave), 10,15.x (Catalina), 11,x (Big Sur), 12,x (Monterey), 13,x (Ventura) et 14,x (Sonoma)

- SUSE Linux Enterprise Server (SLES) 12 et 15
- Ubuntu Server 16.04, 18.04, 20.04 et 22.04
- Les AMIs Windows Server 2008-2012 R2 publiées en novembre 2016 ou après
- Windows Server 2016, 2019 et 2022

> **Note**
>
> SSM Agentpeut être préinstallé sur des appareils AWS gérés AMIs qui ne figurent pas dans cette liste. Cela indique généralement que le système d'exploitation n'est pas entièrement pris en charge par toutes les fonctionnalités de Systems Manager.
> SSM Agentpeut également être préinstallé s'il se AMIs trouve dans AWS Marketplace ou dans le AMIs référentiel communautaire, mais il AWS ne les AMIs prend pas en charge.

Vérifier le statut de SSM Agent

Selon la date d'initialisation, il se peut que SSM Agent ne soit pas préinstallé sur une instance créée à partir d'une AMI figurant dans la liste précédente. Il est également possible que l'agent soit préinstallé sur une instance, mais que celui-ci ne soit pas en cours d'exécution. Par conséquent, nous vous recommandons de vérifier le statut de SSM Agent avant d'essayer d'utiliser Systems Manager sur une instance pour la première fois.

Utilisez la procédure suivante pour vérifier que SSM Agent est installé et exécuté sur une instance. Si vous constatez que l'agent n'est pas installé, procédez manuellement à l'installation sur les instances Linux, macOS et Windows Server.

Pour vérifier l'installation de SSM Agent sur une instance

1. Après avoir lancé une nouvelle instance, patientez quelques minutes pendant son initialisation.
2. Connectez-vous à l'instance en utilisant votre méthode préférée. Par exemple, vous pouvez utiliser SSH pour vous connecter à des instances Linux ou utiliser le Bureau à distance pour vous connecter à des instances Windows Server.
3. Vérifiez l'état de l'SSM Agent en exécutant la commande correspondant au type de système d'exploitation de votre instance.

Système d'exploitation	Command
Amazon Linux 1	`sudo status amazon-ssm-agent`
Amazon Linux 2 et Amazon Linux 2023	`sudo systemctl status amazon-ssm-agent`
macOS	Il n'existe aucune commande permettant de vérifier l'état de l'SSM Agent sur macOS. Vous pouvez vérifier l'état en localisant et en évaluant le fichier journal de l'agent `/var/log/amazon/ssm/amazon-ssm-agent.log` .
SUSE Linux Enterprise Server	`sudo systemctl status amazon-ssm-agent`
Ubuntu Server (32 bits)	`sudo status amazon-ssm-agent`
Ubuntu Server (64 bits – Deb)	`sudo systemctl status amazon-ssm-agent`
Ubuntu Server (64 bits – Snap)	`sudo systemctl status snap.amazon-ssm-agent.amazon-ssm-agent.service`
Windows Server	`Get-Service AmazonSSMAgent`

> **Tip**
>
> Pour afficher les commandes de vérification de l'état de l'SSM Agent sur tous les types de systèmes d'exploitation pris en charge par Systems Manager, veuillez consulter Vérification du statut de l'SSM Agent et démarrage de l'agent.

4. Évaluez la sortie de la commande pour connaître l'état de l'SSM Agent.

   État : installé et en cours d'exécution

Dans la plupart des cas, la sortie de commande indique que l'agent est installé et en cours d'exécution.

L'exemple suivant montre que l'SSM Agent est installé et en cours d'exécution sur une instance Amazon Linux 2.

```
amazon-ssm-agent.service - amazon-ssm-agent
Loaded: loaded (/usr/lib/systemd/system/amazon-ssm-agent.service; enabled; vendor
 preset: enabled)
Active: active (running) since Wed 2021-10-20 19:09:29 UTC; 4min 6s ago
--truncated--
```

L'exemple suivant montre que l'SSM Agent est installé et en cours d'exécution sur une instance Windows Server.

```
Status   Name               DisplayName
------   ----               -----------
Running  AmazonSSMAgent     Amazon SSM Agent
```

État : installé mais pas en cours d'exécution

Dans certains cas, la sortie de commande indique que l'agent est installé, mais qu'il n'est pas en cours d'exécution.

L'exemple suivant montre que l'SSM Agent est installé, mais qu'il n'est pas en cours d'exécution sur une instance Amazon Linux 2.

```
amazon-ssm-agent.service - amazon-ssm-agent
Loaded: loaded (/usr/lib/systemd/system/amazon-ssm-agent.service; enabled; vendor
 preset: enabled)
Active: inactive (dead) since Wed 2021-10-20 22:16:41 UTC; 18s ago
--truncated--
```

L'exemple suivant montre que l'SSM Agent est installé, mais qu'il n'est pas en cours d'exécution sur une instance Windows Server.

```
Status   Name               DisplayName
```

```
------   ----               -----------
Stopped  AmazonSSMAgent     Amazon SSM Agent
```

Si l'agent est installé, mais n'est pas en cours d'exécution, vous pouvez l'activer manuellement à l'aide des commandes correspondant au type de système d'exploitation de votre instance.

Système d'exploitation	Command
Amazon Linux 1	`sudo start amazon-ssm-agent`
Amazon Linux 2 et Amazon Linux 2023	`sudo systemctl enable amazon-ssm-agent` `sudo systemctl start amazon-ssm-agent`
macOS	`sudo launchctl load -w /Library/LaunchDaemons/com.amazon.aws.ssm.plist` `sudo launchctl start com.amazon.aws.ssm`
SUSE Linux Enterprise Server	`sudo systemctl enable amazon-ssm-agent` `sudo systemctl start amazon-ssm-agent`
Ubuntu Server (32 bits)	`sudo start amazon-ssm-agent`
Ubuntu Server (64 bits – Deb)	`sudo systemctl enable amazon-ssm-agent` `sudo systemctl start amazon-ssm-agent`
Ubuntu Server (64 bits – Snap)	`sudo snap start amazon-ssm-agent`

Système d'exploitation	Command
Windows Server	Exécutez la commande suivante dans PowerShell. `Start-Service AmazonSSMAgent`

État : non installé

Dans certains cas, la sortie de commande indique que l'agent n'est pas installé.

L'exemple suivant montre que l'SSM Agent n'est pas installé sur une instance Amazon Linux 2.

```
Unit amazon-ssm-agent.service could not be found.
```

L'exemple suivant montre que l'SSM Agent n'est pas installé sur une instance Windows Server.

```
Get-Service : Cannot find any service with service name 'AmazonSSMAgent'.
--truncated--
```

Si l'agent n'est pas installé, vous pouvez l'installer manuellement à l'aide de la procédure correspondant à votre type de système d'exploitation :

- Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Linux
- Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour macOS
- Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Windows Server

Utilisation de SSM Agent sur des instances EC2 pour Linux

AWS Systems Manager Agent (SSM Agent) traite les demandes de Systems Manager et configure votre machine comme indiqué dans la demande. Utilisez les procédures des rubriques suivantes pour installer, configurer ou désinstaller SSM Agent sur les systèmes d'exploitation Linux.

Rubriques

- Vérification de la signature de SSM Agent
- Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Linux

- Configuration SSM Agent pour utiliser un proxy sur les nœuds Linux

Vérification de la signature de SSM Agent

Les packages d'installation deb et rpm d' AWS Systems Manager Agent (SSM Agent) pour les instances Linux sont signés cryptographiquement. Vous pouvez utiliser une clé publique pour vérifier que le package de l'agent est l'archive originale non modifiée. En cas de dommage ou d'altération des fichiers, la vérification échoue. Vous pouvez vérifier la signature du package d'installation avec RPM ou GPG. Les informations suivantes sont pour les versions 3.1.1141.0 ou ultérieures de SSM Agent.

> **Important**
>
> La clé publique présentée plus loin dans cette rubrique expire le 17/02/2025 (17 février 2025). Systems Manager publiera une nouvelle clé publique dans cette rubrique avant l'expiration de l'ancienne clé. Nous vous recommandons de vous abonner au flux RSS pour ce sujet afin de recevoir une notification lorsque la nouvelle clé est disponible.

Pour trouver le bon fichier SIGNATURE pour l'architecture et le système d'exploitation de votre instance, veuillez consulter le tableau suivant.

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

Architecture	Système d'exploitation	URL du fichier SIGNATURE	Nom du fichier de téléchargement de l'agent
x86_64	AlmaLinux, Amazon Linux 1, Amazon Linux 2, Amazon Linux 2023, CentOS, CentOS Stream,,,	`https://s3.`*`region`*`.amazonaws.com/amazon-ssm-`*`region`*`/latest/linux_amd64/`	`amazon-ssm-agent.rpm`

Architecture	Système d'exploitation	URL du fichier SIGNATURE	Nom du fichier de téléchargement de l'agent
	RHEL Oracle Linux Rocky Linux SLES	`amazon-ssm-agent.rpm.sig` `https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/linux_amd64/amazon-ssm-agent.rpm.sig`	
x86_64	Debian Server, Ubuntu Server	`https://s3.`*`region`*`.amazonaws.com/amazon-ssm-`*`region`*`/latest/debian_amd64/amazon-ssm-agent.deb.sig` `https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/debian_amd64/amazon-ssm-agent.deb.sig`	`amazon-ssm-agent.deb`

Architecture	Système d'exploitation	URL du fichier SIGNATURE	Nom du fichier de téléchargement de l'agent
x86	Amazon Linux 1, Amazon Linux 2, Amazon Linux 2023, CentOS, RHEL	`https://s3.region.amazonaws.com/amazon-ssm-region/latest/linux_386/amazon-ssm-agent.rpm.sig` `https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/linux_386/amazon-ssm-agent.rpm.sig`	`amazon-ssm-agent.rpm`

Architecture	Système d'exploitation	URL du fichier SIGNATURE	Nom du fichier de téléchargement de l'agent
x86	Ubuntu Server	`https://s3.region.amazonaws.com/amazon-ssm-region/latest/debian_386/amazon-ssm-agent.deb.sig` `https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/debian_386/amazon-ssm-agent.deb.sig`	`amazon-ssm-agent.deb`

Architecture	Système d'exploitation	URL du fichier SIGNATURE	Nom du fichier de téléchargement de l'agent
ARM64	Amazon Linux 1, Amazon Linux 2, Amazon Linux 2023, CentOS, RHEL	`https://s3.region.amazonaws.com/amazon-ssm-region/latest/linux_arm64/amazon-ssm-agent.rpm.sig` `https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/linux_arm64/amazon-ssm-agent.rpm.sig`	`amazon-ssm-agent.rpm`

Avant de commencer

Avant de vérifier la signature deSSM Agent, vous devez télécharger le package d'agent approprié pour votre système d'exploitation. Par exemple, `https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/linux_arm64/amazon-ssm-agent.rpm`. Pour plus d'informations sur le téléchargement de SSM Agent packages, consultezInstallation et désinstallation manuelles SSM Agent sur les instances EC2 pour Linux.

GPG

Pour vérifier le package de l'SSM Agent sur un serveur Linux

1. Copiez la clé publique suivante et enregistrez-la dans un fichier appelé `amazon-ssm-agent.gpg`.

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG v2.0.22 (GNU/Linux)

mQENBGTtIoIBCAD2M1aoGIE0FXynAHM/jtuvdAVVaX3Q4ZejTqrX+Jq8ElAMhxyO
GzHu2CDtCYxtVxXK3unptLVt2kGgJwNbhYC393jDeZx5dCda4Nk2YXX1UK3P461i
axuuXRzMYvfM4RZn+7bJTu635tA07q9Xm6MGD4TCTvsjBfViOxbrxOg5ozWbJdSw
fSR8MwUrRfmFpAefRlYfCEuZ8FHywa9U6jLeWt2O/kqrZliJOAGjGzXtB7EZkqKb
faCCxikjjvhF1awdEqSK4DQorC/OvQc4I5kP5y2CJbtXvXO73QH2yE75JMDIIx9x
rOsIRUoSfK3UrWaOVuAnEEn5ueKzZNqGG1J1ABEBAAG0J1NTTSBBZ2VudCA8c3Nt
LWFnZW50LXNpZ25lckBhbWF6b24uY29tPokBPwQTAQIAKQUCZO0iggIbLwUJAsaY
gAcLCQgHAwIBBhUIAgkKCwQWAgMBAh4BAheAAAoJELwfSVyX3QTt+icH/A//tJsW
I+7Ay8FGJh8dJPNy++HIBjVSFdGNJFWNbw1Z8uZcazHEcUCH3FhW4CLQLTZ3OVPz
qvFwzDtRDVIN/Y9EGDhLMFvimrE+/z4olWsJ5DANf6BnX8I5UNIcRt5d8SWH1BEJ
2FWIBZFgKyTDI6XzRC5x4ahtgpOVAGeeKDehs+wh6Ga4W0/K4GsviP1Kyr+Ic2br
NAIq0q0IHyN1q9zam3Y0+jKwEuNmTj+Bjyzshyv/X8S0JWWoXJhkexkOvWeBYNNt
5wI4QcSteyfIzp6KlQF8q11Hzz9D9WaPfcBEYyhq7vLEARobkbQMBzpkmaZua241
0RaWG50HRvrgm4aJAhwEEAECAAYFAmTtIoMACgkQfdCXo9rX9fwwqBAAzkTgYJ38
sWgxpn7Ux/81F2BWR1sVkmP79i++fXyJlKI8xtcJFQZhzeUos69KBUCy7mgx5bYU
P7NA5o9DUbwz/QS0i1Cqm4+jtFlX0MXe4FikXcqfDPnnzN8mVB2H+fa43iHR1PuH
GgUWuNdxzSoIYRmLZXWmeN5YXPcmixlhLzcE2TOQn1mOKcu2fKdLtBQ8KiEkmjiu
naoLxnUcyk1zMhaha+LzEkQdOyasix0ggylN2ViWVnlmfy0niuXDxW0qZWPdLStF
OODiX3iqGmkH3rDfy6nvxxBR4GIs+MGD72fpWzzrINDgkGI2i2t1+0AX/mps3aTy
+ftlgrim8stYWB58XXDAb0vad06sNye5/zDzfr0I9HupJrTzFhaYJQjWPaSlINto
LDJnBXohiUIPRYRcy/k012oFHDWZHT3H6CyjK9UD5UlxA9H7dsJurANs6FOVRe+7
34uJyxDZ/W7zLG4AVG0zxibrUSoaJxwcOjVPVsQAlrwG/GTs7tcAccsJqbJ1Py/w
9AgJl8VU2qc8POsHNXk348gjP7C8PDnGMpZFzr9f5INctRushpiv7onX+aWJVX7T
n2uX/TP3LCyH/MsrNJrJOQnMYFRLQitciP0E+F+eA3v9CY6mDuyb8JSx5HuGGUsG
S4bKBOcA8vimEpwPoT8CE7fdsZ3Qkwdu+pw=
=zr5w
-----END PGP PUBLIC KEY BLOCK-----
```

2. Importez la clé publique dans votre porte-clés et notez la valeur de clé renvoyée.

```
gpg --import amazon-ssm-agent.gpg
```

3. Vérifiez l'empreinte digitale. Veillez à remplacer *key-value (valeur de clé)* par la valeur de l'étape précédente. Nous vous recommandons d'utiliser GPG pour vérifier l'empreinte digitale, même si vous utilisez RPM pour vérifier le package d'installation.

```
gpg --fingerprint key-value
```

La sortie générée lors de l'exécution de cette commande est semblable à ce qui suit.

```
pub   2048R/97DD04ED 2023-08-28 [expires: 2025-02-17]
      Key fingerprint = DE92 C7DA 3E56 E923 31D6  2A36 BC1F 495C 97DD 04ED
uid                  SSM Agent <ssm-agent-signer@amazon.com>
```

L'empreinte digitale doit correspondre à ce qui suit.

```
DE92 C7DA 3E56 E923 31D6 2A36 BC1F 495C 97DD 04ED
```

Si l'empreinte digitale ne correspond pas, n'installez pas l'agent. Contacter AWS Support.

4. Si vous ne l'avez pas déjà fait, téléchargez le fichier SIGNATURE en fonction de l'architecture et du système d'exploitation de votre instance.
5. Vérifiez la signature du package d'installation. Assurez-vous de remplacer le nom du *`fichier de signature`* et *`agent-download-filename`*les valeurs que vous avez spécifiées lors du téléchargement du fichier de signature et de l'agent, comme indiqué dans le tableau ci-dessus dans cette rubrique.

```
gpg --verify signature-filename agent-download-filename
```

Par exemple, pour l'architecture x86_64 sur Amazon Linux 2 :

```
gpg --verify amazon-ssm-agent.rpm.sig amazon-ssm-agent.rpm
```

La sortie générée lors de l'exécution de cette commande est semblable à ce qui suit.

```
gpg: Signature made Thu 31 Aug 2023 07:46:49 PM UTC using RSA key ID 97DD04ED
gpg: Good signature from "SSM Agent <ssm-agent-signer@amazon.com>"
gpg: WARNING: This key is not certified with a trusted signature!
gpg:     There is no indication that the signature belongs to the owner.
Primary key fingerprint: DE92 C7DA 3E56 E923 31D6 2A36 BC1F 495C 97DD 04ED
```

Si le résultat inclut l'expression `BAD signature`, vérifiez si vous avez effectué la procédure correctement. Si cette réponse persiste, contactez l'agent AWS Support et ne l'installez pas. Le message d'avertissement relatif à l'approbation ne signifie pas que la signature n'est pas valide, mais seulement que vous n'avez pas vérifié la clé publique. Une clé est uniquement approuvée si vous ou une personne de confiance l'a signée. Si la sortie inclut la

phrase `Can't check signature: No public key`, vérifiez que vous avez téléchargé la version 3.1.1141.0 de SSM Agent ou une version ultérieure.

RPM

Pour vérifier le package de l'SSM Agent sur un serveur Linux

1. Copiez la clé publique suivante et enregistrez-la dans un fichier appelé `amazon-ssm-agent.gpg`.

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG v2.0.22 (GNU/Linux)

mQENBGTtIoIBCAD2M1aoGIE0FXynAHM/jtuvdAVVaX3Q4ZejTqrX+Jq8ElAMhxyO
GzHu2CDtCYxtVxXK3unptLVt2kGgJwNbhYC393jDeZx5dCda4Nk2YXX1UK3P461i
axuuXRzMYvfM4RZn+7bJTu635tA07q9Xm6MGD4TCTvsjBfViOxbrxOg5ozWbJdSw
fSR8MwUrRfmFpAefRlYfCEuZ8FHywa9U6jLeWt2O/kqrZliJOAGjGzXtB7EZkqKb
faCCxikjjvhF1awdEqSK4DQorC/OvQc4I5kP5y2CJbtXvXO73QH2yE75JMDIIx9x
rOsIRUoSfK3UrWaOVuAnEEn5ueKzZNqGG1J1ABEBAAG0J1NTTSBBZ2VudCA8c3Nt
LWFnZW50LXNpZ25lckBhbWF6b24uY29tPokBPwQTAQIAKQUCZO0iggIbLwUJAsaY
gAcLCQgHAwIBBhUIAgkKCwQWAgMBAh4BAheAAAoJELwfSVyX3QTt+icH/A//tJsW
I+7Ay8FGJh8dJPNy++HIBjVSFdGNJFWNbw1Z8uZcazHEcUCH3FhW4CLQLTZ3OVPz
qvFwzDtRDVIN/Y9EGDhLMFvimrE+/z4olWsJ5DANf6BnX8I5UNIcRt5d8SWH1BEJ
2FWIBZFgKyTDI6XzRC5x4ahtgpOVAGeeKDehs+wh6Ga4W0/K4GsviP1Kyr+Ic2br
NAIq0q0IHyN1q9zam3Y0+jKwEuNmTj+Bjyzshyv/X8S0JWWoXJhkexkOvWeBYNNt
5wI4QcSteyfIzp6KlQF8q11Hzz9D9WaPfcBEYyhq7vLEARobkbQMBzpkmaZua241
0RaWG50HRvrgm4aJAhwEEAECAAYFAmTtIoMACgkQfdCXo9rX9fwwqBAAzkTgYJ38
sWgxpn7Ux/81F2BWR1sVkmP79i++fXyJlKI8xtcJFQZhzeUos69KBUCy7mgx5bYU
P7NA5o9DUbwz/QS0i1Cqm4+jtFlX0MXe4FikXcqfDPnnzN8mVB2H+fa43iHR1PuH
GgUWuNdxzSoIYRmLZXWmeN5YXPcmixlhLzcE2TOQn1mOKcu2fKdLtBQ8KiEkmjiu
naoLxnUcyk1zMhaha+LzEkQdOyasix0ggylN2ViWVnlmfy0niuXDxW0qZWPdLStF
OODiX3iqGmkH3rDfy6nvxxBR4GIs+MGD72fpWzzrINDgkGI2i2t1+0AX/mps3aTy
+ftlgrim8stYWB58XXDAb0vad06sNye5/zDzfr0I9HupJrTzFhaYJQjWPaSlINto
LDJnBXohiUIPRYRcy/k012oFHDWZHT3H6CyjK9UD5UlxA9H7dsJurANs6FOVRe+7
34uJyxDZ/W7zLG4AVG0zxibrUSoaJxwcOjVPVsQAlrwG/GTs7tcAccsJqbJ1Py/w
9AgJl8VU2qc8POsHNXk348gjP7C8PDnGMpZFzr9f5INctRushpiv7onX+aWJVX7T
n2uX/TP3LCyH/MsrNJrJOQnMYFRLQitciP0E+F+eA3v9CY6mDuyb8JSx5HuGGUsG
S4bKBOcA8vimEpwPoT8CE7fdsZ3Qkwdu+pw=
=zr5w
-----END PGP PUBLIC KEY BLOCK-----
```

2. Importez la clé publique dans votre porte-clés et notez la valeur de clé renvoyée.

```
rpm --import amazon-ssm-agent.gpg
```

3. Vérifiez l'empreinte digitale. Veillez à remplacer *key-value (valeur de clé)* par la valeur de l'étape précédente. Nous vous recommandons d'utiliser GPG pour vérifier l'empreinte digitale, même si vous utilisez RPM pour vérifier le package d'installation.

   ```
   gpg --fingerprint key-value
   ```

   La sortie générée lors de l'exécution de cette commande est semblable à ce qui suit.

   ```
   pub   2048R/97DD04ED 2023-08-28 [expires: 2025-02-17]
         Key fingerprint = DE92 C7DA 3E56 E923 31D6  2A36 BC1F 495C 97DD 04ED
   uid                  SSM Agent <ssm-agent-signer@amazon.com>
   ```

   L'empreinte digitale doit correspondre à ce qui suit.

   ```
   DE92 C7DA 3E56 E923 31D6 2A36 BC1F 495C 97DD 04ED
   ```

   Si l'empreinte digitale ne correspond pas, n'installez pas l'agent. Contacter AWS Support.

4. Vérifiez la signature du package d'installation. Assurez-vous de remplacer le nom du *fichier de signature* et *agent-download-filename*les valeurs que vous avez spécifiées lors du téléchargement du fichier de signature et de l'agent, comme indiqué dans le tableau ci-dessus dans cette rubrique.

   ```
   rpm --checksig signature-filename agent-download-filename
   ```

   Par exemple, pour l'architecture x86_64 sur Amazon Linux 2 :

   ```
   rpm --checksig amazon-ssm-agent.rpm.sig amazon-ssm-agent.rpm
   ```

   La sortie générée lors de l'exécution de cette commande est semblable à ce qui suit.

   ```
   amazon-ssm-agent-3.1.1141.0-1.amzn2.x86_64.rpm: rsa sha1 (md5) pgp md5 OK
   ```

   Si la sortie ne contient pas pgp et que vous avez importé la clé publique, alors l'agent n'est pas signé. Si la sortie contient la phrase `NOT OK (MISSING KEYS: (MD5)` *key-id*), vérifiez si vous avez suivi la procédure correctement et vérifiez que vous avez téléchargé

la version 3.1.1141.0 de SSM Agent ou une version ultérieure. Si cette réponse persiste, contactez l'agent AWS Support et ne l'installez pas.

Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Linux

Avant d'installer manuellement AWS Systems Manager Agent (SSM Agent) sur un système d'exploitation Linux Amazon Elastic Compute Cloud (Amazon EC2), consultez les informations suivantes.

URL du fichier d'installation de SSM Agent

Vous pouvez accéder aux fichiers d'installation SSM Agent qui sont stockés dans n'importe quelle publicité Région AWS. Nous fournissons également des fichiers d'installation dans un compartiment Amazon Simple Storage Service (Amazon S3) disponible dans le monde entier que vous pouvez utiliser comme source secondaire ou source de sauvegarde de fichiers.

Si vous installez manuellement l'agent sur une ou deux instances, vous pouvez utiliser les commandes des procédures d'installation rapide que nous fournissons pour gagner du temps. Les commandes fournies dans ces procédures peuvent également être transmises aux instances Amazon EC2 sous forme de scripts via les données utilisateurs.

Si vous créez un script ou un modèle à utiliser pour installer l'agent sur plusieurs instances, nous vous recommandons d'utiliser les fichiers d'installation dans ou à proximité d'une Région AWS où vous vous trouvez. Pour les installations en lot, cela peut augmenter la vitesse de vos téléchargements et réduire la latence. Dans ce cas, nous vous conseillons d'utiliser les procédures Créer des commandes d'installation personnalisées dans les rubriques d'installation.

Amazon Machine Images avec l'agent préinstallé

SSM Agentest préinstallé sur some Amazon Machine Images (AMIs) fourni par AWS. Pour plus d'informations, veuillez consulter Rechercher AMIs avec le SSM Agent préinstallé.

Installation sur d'autres types de machines

Si vous devez installer l'agent sur un serveur local ou une machine virtuelle (VM) afin de pouvoir l'utiliser avec Systems Manager, reportez-vous à la section How to install the SSM Agent on hybrid Linux nodes. Pour plus d'informations sur l'installation de l'agent sur les appareils de périphérie, consultez la rubrique Gestion des appareils de pointe avec Systems Manager.

Maintien de l'agent à jour

Une nouvelle version de SSM Agent est lancée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Le fait de ne pas utiliser la dernière version de l'agent peut empêcher votre nœud géré d'utiliser diverses capacités et fonctionnalités de Systems Manager. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir SSM Agent à jour sur vos machines. Pour plus d'informations, consultez Automatisation des mises à jour de l'SSM Agent. Abonnez-vous à la page des notes de SSM Agent publication GitHub pour recevoir des notifications concernant les SSM Agent mises à jour.

Choisir votre système d'exploitation

Pour afficher la procédure d'installation manuelle de SSM Agent sur le système d'exploitation spécifié, choisissez un lien dans la liste suivante :

> **Note**
>
> Pour obtenir la liste des versions prises en charge de chacun des systèmes d'exploitation suivants, veuillez consulter Systèmes d'exploitation pris en charge pour Systems Manager.

- AlmaLinux
- Amazon Linux 2 et Amazon Linux 2023
- Amazon Linux 1 1
- CentOS
- CentOS Stream
- Debian Server
- Oracle Linux
- Red Hat Enterprise Linux
- Rocky Linux
- SUSE Linux Enterprise Server
- Ubuntu Server

Désinstallation de SSM Agent sur les instances Linux

Utilisez le gestionnaire de packages de votre système d'exploitation pour procéder à la désinstallation SSM Agent des instances Linux. Selon le système d'exploitation, la commande de désinstallation sera similaire à l'exemple de commande suivant :

```
sudo dpkg -r amazon-ssm-agent
```

Installation manuelle de SSM Agent sur les instances AlmaLinux

Utilisez les informations de cette section pour vous aider à installer ou à réinstaller manuellement SSM Agent sur une AlmaLinux instance.

Avant de commencer

Avant de procéder à l'installation SSM Agent sur une AlmaLinux instance, notez les points suivants :

- Assurez-vous que Python 3 est installé sur votre AlmaLinux instance. Ceci est nécessaire pour que SSM Agent fonctionne correctement.
- Pour obtenir des informations importantes qui s'appliquent à l'installation de SSM Agent sur tous les systèmes d'exploitation Linux, consultez la rubrique Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Linux.

Rubriques

- Commandes d'installation rapides pour SSM Agent activer AlmaLinux
- Créez des commandes d'installation d'agents personnalisées pour AlmaLinux votre région

Commandes d'installation rapides pour SSM Agent activer AlmaLinux

Suivez ces étapes pour installer manuellement SSM Agent sur une seule instance. Cette procédure utilise les fichiers d'installation disponibles dans le monde entier.

Avant de commencer

Avant de procéder à l'installation SSM Agent sur une AlmaLinux instance, notez les points suivants :

- Assurez-vous que Python 3 est installé sur votre AlmaLinux instance. Ceci est nécessaire pour que SSM Agent fonctionne correctement.

À installer SSM Agent sur AlmaLinux

1. Connectez-vous à votre AlmaLinux instance à l'aide de votre méthode préférée, telle que SSH.
2. Copiez la commande de l'architecture de votre instance et exécutez-la sur l'instance.

> **Note**
>
> Même si les URL des commandes suivantes incluent un `ec2-downloads-windows` répertoire, il s'agit des fichiers d'installation globale appropriés pour AlmaLinux.

Instances x86_64

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

Instances ARM64

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

3. (Recommandé) Exécutez la commande suivante pour vérifier que l'agent est en cours d'exécution.

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

Dans la plupart des cas, la commande signale que l'agent est en cours d'exécution, comme illustré dans l'exemple suivant.

```
# amazon-ssm-agent.service - amazon-ssm-agent
   Loaded: loaded (/etc/systemd/system/amazon-ssm-agent.service; enabled; vendo>
   Active: active (running) since Tue 2022-04-19 16:40:41 UTC; 9s ago
 Main PID: 4898 (amazon-ssm-agen)
    Tasks: 14 (limit: 4821)
   Memory: 34.6M
   CGroup: /system.slice/amazon-ssm-agent.service
           ##4898 /usr/bin/amazon-ssm-agent
           ##4954 /usr/bin/ssm-agent-worker
            --truncated--
```

Dans de rares cas, la commande signale que l'agent est installé, mais qu'il n'est pas en cours d'exécution, comme illustré dans l'exemple suivant.

```
# amazon-ssm-agent.service - amazon-ssm-agent
   Loaded: loaded (/etc/systemd/system/amazon-ssm-agent.service; enabled; vendo>
   Active: inactive (dead) since Tue 2022-04-19 16:42:05 UTC; 2s ago
            --truncated--
```

Pour activer l'agent dans ces cas-là, exécutez les commandes suivantes.

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

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

Créez des commandes d'installation d'agents personnalisées pour AlmaLinux votre région

Lorsque vous installez SSM Agent sur plusieurs instances utilisant un script ou un modèle, nous vous recommandons d'utiliser les fichiers d'installation stockés dans l' Région AWS dans laquelle vous travaillez.

S'il s'agit des commandes suivantes, nous fournissons des exemples qui utilisent un compartiment S3 accessible au public dans la région USA Est (Ohio) (`us-east-2`).

> **Tip**
>
> Vous pouvez également remplacer une URL globale dans la procédure Commandes d'installation rapides pour SSM Agent activer AlmaLinux plus avant dans cette rubrique, avec une URL régionale personnalisée que vous créez.

Dans la commande ci-après, remplacez *`region`* (région) par vos propres informations. Pour obtenir une liste des valeurs *`region`* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la Référence générale d'Amazon Web Services.

x86_64

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

Consultez l'exemple suivant.

```
sudo dnf install -y https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/
linux_amd64/amazon-ssm-agent.rpm
```

ARM64

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

Consultez l'exemple suivant.

```
sudo dnf install -y https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/
linux_arm64/amazon-ssm-agent.rpm
```

Installation manuelle de SSM Agent sur les instances Amazon Linux 2 et Amazon Linux 2023

> **Important**
>
> Cette rubrique fournit des commandes à utiliser SSM Agent sur les instances Amazon Linux 2 et Amazon Linux 2023. Certaines de ces commandes ne sont pas prises en charge sur les instances Amazon Linux 1. Avant de continuer, assurez-vous que vous regardez bien la rubrique correspondant à votre type d'instance. Pour les commandes à exécuter sur les instances Amazon Linux 1, consultezInstallation manuelle SSM Agent sur des instances Amazon Linux 1.

Dans la plupart des cas, les Amazon Machine Images (AMIs) pour Amazon Linux 2 et Amazon Linux 2023 fournis par sont AWS fournis avec AWS Systems Manager Agent (SSM Agent) préinstallé par défaut. Pour plus d'informations, consultez Rechercher AMIs avec le SSM Agent préinstallé.

Dans le cas où SSM Agent n'est pas préinstallé sur une nouvelle instance Amazon Linux 2 ou Amazon Linux 2023, ou si vous devez réinstaller manuellement l'agent, utilisez les informations de cette page pour vous aider.

Avant de commencer

Avant d'installer SSM Agent sur une instance Amazon Linux 2 ou Amazon Linux 2023, notez ce qui suit :

- Pour obtenir des informations importantes qui s'appliquent à l'installation de SSM Agent sur tous les systèmes d'exploitation Linux, consultez la rubrique Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Linux.
- Si vous utilisez une commande `yum` pour mettre à jour l'SSM Agent sur un nœud géré après que l'agent a été installé ou mis à jour en utilisant le document SSM `AWS-UpdateSSMAgent`, le message suivant peut s'afficher : « Warning: RPMDB altered outside of yum (Avertissement : RPMDB modifié en dehors de yum) ». Ce message est prévu pour s'afficher. Il peut être ignoré sans risque.

Rubriques

- Commandes d'installation rapide pour SSM Agent sur Amazon Linux 2 ou Amazon Linux 2023
- Création de commandes d'installation d'agent personnalisées pour Amazon Linux 2 ou Amazon Linux 2023 dans votre région

Commandes d'installation rapide pour SSM Agent sur Amazon Linux 2 ou Amazon Linux 2023

Suivez ces étapes pour installer manuellement SSM Agent sur une seule instance. Cette procédure utilise les fichiers d'installation disponibles dans le monde entier.

Pour installer SSM Agent sur Amazon Linux 2 ou Amazon Linux 2023 à l'aide de commandes de copier-coller rapides

1. Connectez-vous à votre instance Amazon Linux 2 ou Amazon Linux 2023 à l'aide de votre méthode préférée, telle que SSH.
2. Copiez la commande de l'architecture de votre instance et exécutez-la sur l'instance.

> **Note**
>
> Même si les URL des commandes suivantes incluent un répertoire `ec2-downloads-windows`, il s'agit des fichiers d'installation globaux corrects pour Amazon Linux 2 et Amazon Linux 2023.

x86_64

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

ARM64

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

3. (Recommandé) Exécutez la commande suivante pour vérifier que l'agent est en cours d'exécution.

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

Dans la plupart des cas, la commande signale que l'agent est en cours d'exécution, comme illustré dans l'exemple suivant.

```
amazon-ssm-agent.service - amazon-ssm-agent
Loaded: loaded (/usr/lib/systemd/system/amazon-ssm-agent.service; enabled; vendor
 preset: enabled)
Active: active (running) since Wed 2021-10-20 19:09:29 UTC; 4min 6s ago
            --truncated--
```

Dans de rares cas, la commande signale que l'agent est installé, mais qu'il n'est pas en cours d'exécution, comme illustré dans l'exemple suivant.

```
amazon-ssm-agent.service - amazon-ssm-agent
Loaded: loaded (/usr/lib/systemd/system/amazon-ssm-agent.service; enabled; vendor
 preset: enabled)
Active: inactive (dead) since Wed 2021-10-20 22:16:41 UTC; 18s ago
            --truncated--
```

Dans ces cas, exécutez la commande suivante pour activer l'agent.

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

Création de commandes d'installation d'agent personnalisées pour Amazon Linux 2 ou Amazon Linux 2023 dans votre région

Lorsque vous installez SSM Agent sur plusieurs instances utilisant un script ou un modèle, nous vous recommandons d'utiliser les fichiers d'installation stockés dans l' Région AWS dans laquelle vous travaillez.

S'il s'agit des commandes suivantes, nous fournissons des exemples qui utilisent un compartiment S3 accessible au public dans la région USA Est (Ohio) (`us-east-2`).

> **Tip**
>
> Vous pouvez également remplacer une URL globale dans la procédure Commandes d'installation rapides pour SSM Agent Amazon Linux 1 plus avant dans cette rubrique, avec une URL régionale personnalisée que vous créez.

Dans la commande ci-après, remplacez *`region`* (région) par vos propres informations. Pour obtenir une liste des valeurs *`region`* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la Référence générale d'Amazon Web Services.

x86_64

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

Consultez l'exemple suivant.

```
sudo yum install -y https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/
linux_amd64/amazon-ssm-agent.rpm
```

ARM64

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

Consultez l'exemple suivant.

```
sudo yum install -y https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/
linux_arm64/amazon-ssm-agent.rpm
```

Installation manuelle SSM Agent sur des instances Amazon Linux 1

> **Important**
>
> Amazon Linux 1 a atteint la fin de son support standard le 31 décembre 2020 et a atteint sa fin de vie le 31 décembre 2023, comme annoncé dans la mise à jour sur Amazon Linux AMI end-of-life sur le blog d'AWS actualités. AWS ne fournit plus Amazon Machine Images (AMIs) pour ce système d'exploitation. AWS Systems Manager continue toutefois de fournir un support pour les instances Amazon Linux 1 existantes.
> Cette rubrique fournit des commandes à utiliser SSM Agent sur les instances Amazon Linux 1. Certaines commandes ne sont pas prises en charge sur les instances Amazon Linux 2 et Amazon Linux 2023. Avant de continuer, vérifiez que vous regardez bien la rubrique correspondant à votre type d'instance. Pour savoir quelles commandes exécuter sur les instances Amazon Linux 2 ou Amazon Linux 2023, veuillez consulter la rubrique Installation manuelle de SSM Agent sur les instances Amazon Linux 2 et Amazon Linux 2023.

Dans la plupart des cas, les Amazon Machine Images (AMIs) pour Amazon Linux 1 fournis par sont AWS fournis avec AWS Systems Manager Agent (SSM Agent) préinstallé par défaut. Pour plus d'informations, consultez Rechercher AMIs avec le SSM Agent préinstallé.

Si vous devez réinstaller manuellement l'agent sur Amazon Linux 1, utilisez les informations de cette page pour vous aider.

Avant de commencer

Avant de procéder à l'installation SSM Agent sur une instance Amazon Linux 1, notez ce qui suit :

- Pour obtenir des informations importantes qui s'appliquent à l'installation de SSM Agent sur tous les systèmes d'exploitation Linux, consultez la rubrique Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Linux.
- Si vous utilisez une commande `yum` pour mettre à jour l'SSM Agent sur un nœud géré après que l'agent a été installé ou mis à jour en utilisant le document SSM `AWS-UpdateSSMAgent`, le message suivant peut s'afficher : « Warning: RPMDB altered outside of yum (Avertissement : RPMDB modifié en dehors de yum) ». Ce message est prévu pour s'afficher. Il peut être ignoré sans risque.

Rubriques

- Commandes d'installation rapides pour SSM Agent Amazon Linux 1
- Créez des commandes d'installation d'agents personnalisées pour Amazon Linux 1 dans votre région

Commandes d'installation rapides pour SSM Agent Amazon Linux 1

Suivez ces étapes pour installer manuellement SSM Agent sur une seule instance. Cette procédure utilise les fichiers d'installation disponibles dans le monde entier.

Pour installer SSM Agent sur Amazon Linux 1 à l'aide des commandes rapides de copier-coller

1. Connectez-vous à votre instance Amazon Linux 1 à l'aide de votre méthode préférée, telle que SSH.
2. Copiez la commande de l'architecture de votre instance et exécutez-la sur l'instance.

> **Note**
>
> Même si les URL des commandes suivantes incluent un `ec2-downloads-windows` répertoire, il s'agit des fichiers d'installation globaux corrects pour Amazon Linux 1.

x86_64

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

x86

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

ARM64

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

3. (Recommandé) Exécutez la commande de l'architecture de votre instance pour vérifier que l'agent est en cours d'exécution.

x86_64 et x86

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

ARM64

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

Dans la plupart des cas, la commande indique que l'agent est en cours d'exécution, comme le montrent les exemples suivants.

x86_64 et x86

```
amazon-ssm-agent start/running, process 12345
```

ARM64

```
amazon-ssm-agent.service - amazon-ssm-agent
Loaded: loaded (/usr/lib/systemd/system/amazon-ssm-agent.service; enabled;
 vendor preset: enabled)
Active: active (running) since Wed 2021-10-20 19:09:29 UTC; 4min 6s ago
            --truncated--
```

Dans de rares cas, la commande signale que l'agent est installé, mais qu'il n'est pas en cours d'exécution, comme illustré dans les exemples suivants.

x86_64 et x86

```
amazon-ssm-agent stop/waiting
```

ARM64

```
amazon-ssm-agent.service - amazon-ssm-agent
Loaded: loaded (/usr/lib/systemd/system/amazon-ssm-agent.service; enabled;
 vendor preset: enabled)
Active: inactive (dead) since Wed 2021-10-20 22:16:41 UTC; 18s ago
            --truncated--
```

Pour activer l'agent dans ces cas, exécutez la commande correspondant à l'architecture de votre instance.

x86_64 et x86

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

ARM64

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

Créez des commandes d'installation d'agents personnalisées pour Amazon Linux 1 dans votre région

Lorsque vous installez SSM Agent sur plusieurs instances utilisant un script ou un modèle, nous vous recommandons d'utiliser les fichiers d'installation stockés dans l' Région AWS dans laquelle vous travaillez.

S'il s'agit des commandes suivantes, nous fournissons des exemples qui utilisent un compartiment S3 accessible au public dans la région USA Est (Ohio) (`us-east-2`).

> **Tip**
>
> Vous pouvez également remplacer une URL globale dans la procédure Commandes d'installation rapides pour SSM Agent Amazon Linux 1 plus avant dans cette rubrique, avec une URL régionale personnalisée que vous créez.

Dans la commande ci-après, remplacez *`region`* (région) par vos propres informations. Pour obtenir une liste des valeurs *`region`* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la Référence générale d'Amazon Web Services.

x86_64

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

Consultez l'exemple suivant.

```
sudo yum install -y https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/
linux_amd64/amazon-ssm-agent.rpm
```

x86

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

Consultez l'exemple suivant.

```
sudo yum install -y https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/
linux_386/amazon-ssm-agent.rpm
```

ARM64

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

Consultez l'exemple suivant.

```
sudo yum install -y https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/
linux_arm64/amazon-ssm-agent.rpm
```

Installation manuelle de SSM Agent sur les instances CentOS

Les Amazon Machine Images (AMIs) pour CentOS fournis par ne sont AWS pas fournis avec AWS Systems Manager Agent (SSM Agent) préinstallé par défaut. Pour obtenir une liste des AMIs gérées par AWS sur lesquelles l'agent peut être préinstallé, consultez la rubrique Rechercher AMIs avec le SSM Agent préinstallé.

Utilisez les informations de cette section pour vous aider à installer ou à réinstaller manuellement SSM Agent sur une instance CentOS.

Avant de commencer

Avant d'installer SSM Agent sur une instance CentOS, notez ce qui suit :

- Pour obtenir des informations importantes qui s'appliquent à l'installation de SSM Agent sur tous les systèmes d'exploitation Linux, consultez la rubrique Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Linux.

- Si vous utilisez une commande `yum` pour mettre à jour l'SSM Agent sur un nœud géré après que l'agent a été installé ou mis à jour en utilisant le document SSM `AWS-UpdateSSMAgent`, le message suivant peut s'afficher : « Warning: RPMDB altered outside of yum (Avertissement : RPMDB modifié en dehors de yum) ». Ce message est prévu pour s'afficher. Il peut être ignoré sans risque.

Rubriques

Installation de SSM Agent sur CentOS 8.x

Les Amazon Machine Images (AMIs) pour CentOS 8 fournies par AWS n'incluent pas l'agent AWS Systems Manager (SSM Agent) par défaut. Utilisez les informations de cette page pour vous aider à installer ou à réinstaller l'agent sur les instances CentOS 8.

Avant de commencer

Avant d'installer SSM Agent sur une instance CentOS 8, notez ce qui suit :

- Assurez-vous que Python 2 ou Python 3 est installé sur votre instance CentOS 8. Ceci est nécessaire pour que SSM Agent fonctionne correctement.

Rubriques

Commandes d'installation rapide pour SSM Agent sur CentOS 8

Suivez ces étapes pour installer manuellement SSM Agent sur une seule instance. Cette procédure utilise les fichiers d'installation disponibles dans le monde entier.

Pour installer l'SSM Agent sur CentOS 8.x

1. Connectez-vous à l'instance CentOS 8 à l'aide de votre méthode préférée, telle que SSH.
2. Copiez la commande de l'architecture de votre instance et exécutez-la sur l'instance.

> **Note**
>
> Même si les URL des commandes suivantes incluent un répertoire `ec2-downloads-windows`, il s'agit des fichiers d'installation globaux corrects pour CentOS 8.

x86_64Instances

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

Instances ARM64

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

3. (Recommandé) Exécutez la commande suivante pour vérifier que l'agent est en cours d'exécution.

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

Dans la plupart des cas, la commande signale que l'agent est en cours d'exécution, comme illustré dans l'exemple suivant.

```
# amazon-ssm-agent.service - amazon-ssm-agent
   Loaded: loaded (/etc/systemd/system/amazon-ssm-agent.service; enabled; vendo>
   Active: active (running) since Tue 2022-04-19 15:48:54 UTC; 19s ago
            --truncated--
```

Dans de rares cas, la commande signale que l'agent est installé, mais qu'il n'est pas en cours d'exécution, comme illustré dans l'exemple suivant.

```
# amazon-ssm-agent.service - amazon-ssm-agent
   Loaded: loaded (/etc/systemd/system/amazon-ssm-agent.service; disabled; vend>
   Active: inactive (dead)
            --truncated--
```

Pour activer l'agent dans ces cas-là, exécutez les commandes suivantes.

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

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

Création de commandes d'installation d'agent personnalisées pour CentOS 8 dans votre région

Lorsque vous installez SSM Agent sur plusieurs instances utilisant un script ou un modèle, nous vous recommandons d'utiliser les fichiers d'installation stockés dans l'Région AWS dans laquelle vous travaillez.

S'il s'agit des commandes suivantes, nous fournissons des exemples qui utilisent un compartiment S3 accessible au public dans la région USA Est (Ohio) (`us-east-2`).

> **Tip**
>
> Vous pouvez également remplacer une URL globale dans la procédure Commandes d'installation rapide pour SSM Agent sur CentOS 8 plus avant dans cette rubrique, avec une URL régionale personnalisée que vous créez.

Dans la commande ci-après, remplacez *`region`* (région) par vos propres informations. Pour obtenir une liste des valeurs *`region`* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la Référence générale d'Amazon Web Services.

x86_64

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

Consultez l'exemple suivant.

```
sudo dnf install -y https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/
linux_amd64/amazon-ssm-agent.rpm
```

ARM64

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

Consultez l'exemple suivant.

```
sudo dnf install -y https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/
linux_arm64/amazon-ssm-agent.rpm
```

Installation de SSM Agent sur CentOS 7.x

Les Amazon Machine Images (AMIs) pour CentOS 7 fournies par AWS n'incluent pas l'agent AWS Systems Manager (SSM Agent) par défaut. Utilisez les informations de cette page pour vous aider à installer ou à réinstaller l'agent sur les instances CentOS 7.

Rubriques

- Commandes d'installation rapide pour SSM Agent sur CentOS 7
- Création de commandes d'installation d'agent personnalisées pour CentOS 7 dans votre région

Commandes d'installation rapide pour SSM Agent sur CentOS 7

Suivez ces étapes pour installer manuellement SSM Agent sur une seule instance. Cette procédure utilise les fichiers d'installation disponibles dans le monde entier.

Installer l'SSM Agent sur CentOS 7.x

1. Connectez-vous à l'instance CentOS 7 à l'aide de votre méthode préférée, telle que SSH.
2. Copiez la commande de l'architecture de votre instance et exécutez-la sur l'instance.

> **Note**
>
> Même si les URL des commandes suivantes incluent un répertoire `ec2-downloads-windows`, il s'agit des fichiers d'installation globaux corrects pour CentOS 7.

Instances x86_64

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

Instances ARM64

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

3. (Recommandé) Exécutez la commande suivante pour vérifier que l'agent est en cours d'exécution.

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

Dans la plupart des cas, la commande signale que l'agent est en cours d'exécution, comme illustré dans l'exemple suivant.

```
# amazon-ssm-agent.service - amazon-ssm-agent
   Loaded: loaded (/etc/systemd/system/amazon-ssm-agent.service; enabled; vendor
 preset: disabled)
   Active: active (running) since Tue 2022-04-19 15:57:27 UTC; 6s ago
            --truncated--
```

Dans de rares cas, la commande signale que l'agent est installé, mais qu'il n'est pas en cours d'exécution, comme illustré dans l'exemple suivant.

```
# amazon-ssm-agent.service - amazon-ssm-agent
   Loaded: loaded (/etc/systemd/system/amazon-ssm-agent.service; enabled; vendor
 preset: disabled)
   Active: inactive (dead) since Tue 2022-04-19 15:58:44 UTC; 2s ago
            --truncated--
```

Pour activer l'agent dans ces cas-là, exécutez les commandes suivantes.

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

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

Création de commandes d'installation d'agent personnalisées pour CentOS 7 dans votre région

Lorsque vous installez SSM Agent sur plusieurs instances utilisant un script ou un modèle, nous vous recommandons d'utiliser les fichiers d'installation stockés dans l'Région AWS dans laquelle vous travaillez.

S'il s'agit des commandes suivantes, nous fournissons des exemples qui utilisent un compartiment S3 accessible au public dans la région USA Est (Ohio) (`us-east-2`).

> **Tip**
>
> Vous pouvez également remplacer une URL globale dans la procédure Commandes d'installation rapide pour SSM Agent sur CentOS 7 plus avant dans cette rubrique, avec une URL régionale personnalisée que vous créez.

Dans la commande ci-après, remplacez *`region`* (région) par vos propres informations. Pour obtenir une liste des valeurs *`region`* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la Référence générale d'Amazon Web Services.

x86_64

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

Consultez l'exemple suivant.

```
sudo yum install -y https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/
linux_amd64/amazon-ssm-agent.rpm
```

ARM64

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

Consultez l'exemple suivant.

```
sudo yum install -y https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/
linux_arm64/amazon-ssm-agent.rpm
```

Installation de SSM Agent sur CentOS 6.x

Les Amazon Machine Images (AMIs) pour CentOS 6 fournies par AWS n'incluent pas l'agent AWS Systems Manager (SSM Agent) par défaut. Utilisez les informations de cette page pour vous aider à installer ou à réinstaller l'agent sur les instances CentOS 6.

Rubriques

- Commandes d'installation rapide pour SSM Agent sur CentOS 6
- Création de commandes d'installation d'agent personnalisées pour CentOS 6 dans votre région

Commandes d'installation rapide pour SSM Agent sur CentOS 6

Suivez ces étapes pour installer manuellement SSM Agent sur une seule instance. Cette procédure utilise les fichiers d'installation disponibles dans le monde entier.

Installer l'SSM Agent sur CentOS 6.x

1. Connectez-vous à l'instance CentOS 6 à l'aide de votre méthode préférée, telle que SSH.
2. Copiez la commande de l'architecture de votre instance et exécutez-la sur l'instance.

> **Note**
>
> Même si les URL des commandes suivantes incluent un répertoire `ec2-downloads-windows`, il s'agit des fichiers d'installation globaux corrects pour CentOS 6.
> Les commandes suivantes spécifient le répertoire des versions `3.0.1479.0` au lieu du répertoire `latest`. C'est parce que les versions 3.1 et ultérieures de SSM Agent ne sont pas prises en charge pour CentOS 6.

Instances x86_64

```
sudo yum install -y https://s3.amazonaws.com/ec2-downloads-windows/
SSMAgent/3.0.1479.0/linux_amd64/amazon-ssm-agent.rpm
```

Instances x86

```
sudo yum install -y https://s3.amazonaws.com/ec2-downloads-windows/
SSMAgent/3.0.1479.0/linux_386/amazon-ssm-agent.rpm
```

3. (Recommandé) Exécutez la commande suivante pour vérifier que l'agent est en cours d'exécution.

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

Dans la plupart des cas, la commande signale que l'agent est en cours d'exécution, comme illustré dans l'exemple suivant.

```
amazon-ssm-agent start/running, process 1744
```

Dans de rares cas, la commande signale que l'agent est installé, mais qu'il n'est pas en cours d'exécution, comme illustré dans l'exemple suivant.

```
amazon-ssm-agent stop/waiting
```

Dans ces cas, exécutez la commande suivante pour activer l'agent.

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

Création de commandes d'installation d'agent personnalisées pour CentOS 6 dans votre région

Lorsque vous installez SSM Agent sur plusieurs instances utilisant un script ou un modèle, nous vous recommandons d'utiliser les fichiers d'installation stockés dans l'Région AWS dans laquelle vous travaillez.

S'il s'agit des commandes suivantes, nous fournissons des exemples qui utilisent un compartiment S3 accessible au public dans la région USA Est (Ohio) (`us-east-2`).

> **Tip**
>
> Vous pouvez également remplacer une URL globale dans la procédure Commandes d'installation rapide pour SSM Agent sur CentOS 6 plus avant dans cette rubrique, avec une URL régionale personnalisée que vous créez.

Dans la commande ci-après, remplacez *region* (région) par vos propres informations. Pour obtenir une liste des valeurs *region* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la Référence générale d'Amazon Web Services.

> **Note**
>
> Les commandes suivantes spécifient le répertoire des versions `3.0.1390.0` au lieu du répertoire `latest`. C'est parce que les versions 3.1 et ultérieures de SSM Agent ne sont pas prises en charge pour CentOS 6.

x86_64

```
sudo yum install -y https://s3.region.amazonaws.com/amazon-ssm-region/3.0.1479.0/
linux_amd64/amazon-ssm-agent.rpm
```

Consultez l'exemple suivant.

```
sudo yum install -y https://s3.us-east-2.amazonaws.com/amazon-ssm-us-
east-2/3.0.1479.0/linux_amd64/amazon-ssm-agent.rpm
```

x86

```
sudo yum install -y https://s3.region.amazonaws.com/amazon-ssm-region/3.0.1479.0/
linux_386/amazon-ssm-agent.rpm
```

Consultez l'exemple suivant.

```
sudo yum install -y https://s3.us-east-2.amazonaws.com/amazon-ssm-us-
east-2/3.0.1479.0/linux_386/amazon-ssm-agent.rpm
```

Installation manuelle de SSM Agent sur les instances CentOS Stream

Les Amazon Machine Images (AMIs) pour CentOS Stream cela sont fournis par AWS do not come with AWS Systems Manager Agent (SSM Agent) préinstallé par défaut. Pour obtenir une liste des AMIs gérées par AWS sur lesquelles l'agent peut être préinstallé, consultez la rubrique Rechercher AMIs avec le SSM Agent préinstallé.

Utilisez les informations de cette section pour vous aider à installer ou à réinstaller manuellement SSM Agent sur une instance CentOS Stream.

Avant de commencer

Avant d'installer SSM Agent sur une instance CentOS Stream, notez ce qui suit :

- Pour obtenir des informations importantes qui s'appliquent à l'installation de SSM Agent sur tous les systèmes d'exploitation Linux, consultez la rubrique Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Linux.

Rubriques

- Commandes d'installation rapide pour SSM Agent sur CentOS Stream
- Création de commandes d'installation d'agent personnalisées pour CentOS Stream dans votre région

Commandes d'installation rapide pour SSM Agent sur CentOS Stream

Suivez ces étapes pour installer manuellement SSM Agent sur une seule instance. Cette procédure utilise les fichiers d'installation disponibles dans le monde entier.

Avant de commencer

Avant d'installer SSM Agent sur une instance CentOS Stream, notez ce qui suit :

- Assurez-vous que Python 2 ou Python 3 est installé sur votre instance CentOS Stream 8. Ceci est nécessaire pour que SSM Agent fonctionne correctement.

Pour installer l'SSM Agent sur CentOS Stream

1. Connectez-vous à l'instance CentOS Stream en utilisant votre méthode préférée, telle que SSH.

2. Copiez la commande de l'architecture de votre instance et exécutez-la sur l'instance.

> **Note**
>
> Même si les URL des commandes suivantes incluent un répertoire `ec2-downloads-windows`, il s'agit des fichiers d'installation globaux corrects pour CentOS Stream.

Instances x86_64

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

Instances ARM64

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

3. (Recommandé) Exécutez la commande suivante pour vérifier que l'agent est en cours d'exécution.

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

Dans la plupart des cas, la commande signale que l'agent est en cours d'exécution, comme illustré dans l'exemple suivant.

```
# amazon-ssm-agent.service - amazon-ssm-agent
   Loaded: loaded (/etc/systemd/system/amazon-ssm-agent.service; enabled; vendo>
   Active: active (running) since Tue 2022-04-19 16:40:41 UTC; 9s ago
 Main PID: 4898 (amazon-ssm-agen)
    Tasks: 14 (limit: 4821)
   Memory: 34.6M
   CGroup: /system.slice/amazon-ssm-agent.service
           ##4898 /usr/bin/amazon-ssm-agent
           ##4954 /usr/bin/ssm-agent-worker
            --truncated--
```

Dans de rares cas, la commande signale que l'agent est installé, mais qu'il n'est pas en cours d'exécution, comme illustré dans l'exemple suivant.

```
# amazon-ssm-agent.service - amazon-ssm-agent
   Loaded: loaded (/etc/systemd/system/amazon-ssm-agent.service; enabled; vendo>
   Active: inactive (dead) since Tue 2022-04-19 16:42:05 UTC; 2s ago
            --truncated--
```

Pour activer l'agent dans ces cas-là, exécutez les commandes suivantes.

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

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

Création de commandes d'installation d'agent personnalisées pour CentOS Stream dans votre région

Lorsque vous installez SSM Agent sur plusieurs instances utilisant un script ou un modèle, nous vous recommandons d'utiliser les fichiers d'installation stockés dans l' Région AWS dans laquelle vous travaillez.

S'il s'agit des commandes suivantes, nous fournissons des exemples qui utilisent un compartiment S3 accessible au public dans la région USA Est (Ohio) (`us-east-2`).

> **Tip**
>
> Vous pouvez également remplacer une URL globale dans la procédure Commandes d'installation rapide pour SSM Agent sur CentOS Stream plus avant dans cette rubrique, avec une URL régionale personnalisée que vous créez.

Dans la commande ci-après, remplacez *`region`* (région) par vos propres informations. Pour obtenir une liste des valeurs *`region`* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la Référence générale d'Amazon Web Services.

x86_64

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

Consultez l'exemple suivant.

```
sudo dnf install -y https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/
linux_amd64/amazon-ssm-agent.rpm
```

ARM64

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

Consultez l'exemple suivant.

```
sudo dnf install -y https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/
linux_arm64/amazon-ssm-agent.rpm
```

Installation manuelle de SSM Agent sur les instances Debian Server

Les Amazon Machine Images (AMIs) pour Debian Server cela sont fournis par AWS do not come with AWS Systems Manager Agent (SSM Agent) préinstallé par défaut. Pour obtenir une liste des AMIs gérées par AWS sur lesquelles l'agent peut être préinstallé, consultez la rubrique Rechercher AMIs avec le SSM Agent préinstallé.

Utilisez les informations de cette section pour vous aider à installer ou à réinstaller manuellement SSM Agent sur une instance Debian Server.

Avant de commencer

Avant d'installer SSM Agent sur une instance Debian Server, notez ce qui suit :

- Pour obtenir des informations importantes qui s'appliquent à l'installation de SSM Agent sur tous les systèmes d'exploitation Linux, consultez la rubrique Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Linux.

Rubriques

- Commandes d'installation rapide pour SSM Agent sur Debian Server
- Création de commandes d'installation d'agent personnalisées pour Debian Server dans votre région

Commandes d'installation rapide pour SSM Agent sur Debian Server

Suivez ces étapes pour installer manuellement SSM Agent sur une seule instance. Cette procédure utilise les fichiers d'installation disponibles dans le monde entier.

Pour installer l'SSM Agent sur Debian Server

1. Connectez-vous à l'instance Debian Server en utilisant votre méthode préférée, telle que SSH.
2. Exécutez la commande suivante pour créer un répertoire temporaire sur l'instance.

```
mkdir /tmp/ssm
```

3. Exécutez la commande suivante pour passer dans le répertoire temporaire.

```
cd /tmp/ssm
```

4. Copiez la commande de l'architecture de votre instance et exécutez-la sur l'instance.

> **Note**
>
> Même si les URL des commandes suivantes incluent un répertoire `ec2-downloads-windows`, il s'agit des fichiers d'installation globaux corrects pour Debian Server. Pour Debian Server 8, seule l'architecture x86_64 est prise en charge.

Instances x86_64

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

Instances ARM64

```
wget https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/
debian_arm64/amazon-ssm-agent.deb
```

5. Exécutez la commande suivante.

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

6. (Recommandé) Exécutez la commande suivante pour vérifier que l'agent est en cours d'exécution.

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

Dans la plupart des cas, la commande signale que l'agent est en cours d'exécution, comme illustré dans l'exemple suivant.

```
# amazon-ssm-agent.service - amazon-ssm-agent
   Loaded: loaded (/lib/systemd/system/amazon-ssm-agent.service; enabled; vendor
   Active: active (running) since Tue 2022-04-19 16:25:03 UTC; 4s ago
 Main PID: 628 (amazon-ssm-agen)
   CGroup: /system.slice/amazon-ssm-agent.service
           ##628 /usr/bin/amazon-ssm-agent
           ##650 /usr/bin/ssm-agent-worker
            --truncated--
```

Dans de rares cas, la commande signale que l'agent est installé, mais qu'il n'est pas en cours d'exécution, comme illustré dans l'exemple suivant.

```
# amazon-ssm-agent.service - amazon-ssm-agent
   Loaded: loaded (/lib/systemd/system/amazon-ssm-agent.service; enabled; vendor
   Active: inactive (dead) since Tue 2022-04-19 16:26:30 UTC; 5s ago
 Main PID: 628 (code=exited, status=0/SUCCESS)
            --truncated--
```

Pour activer l'agent dans ces cas-là, exécutez les commandes suivantes.

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

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

Création de commandes d'installation d'agent personnalisées pour Debian Server dans votre région

Lorsque vous installez SSM Agent sur plusieurs instances utilisant un script ou un modèle, nous vous recommandons d'utiliser les fichiers d'installation stockés dans l' Région AWS dans laquelle vous travaillez.

S'il s'agit des commandes suivantes, nous fournissons des exemples qui utilisent un compartiment S3 accessible au public dans la région USA Est (Ohio) (`us-east-2`).

> **Tip**
>
> Vous pouvez également remplacer une URL globale dans la procédure Commandes d'installation rapide pour SSM Agent sur Debian Server plus avant dans cette rubrique, avec une URL régionale personnalisée que vous créez.

Dans la commande ci-après, remplacez *`region`* (région) par vos propres informations. Pour obtenir une liste des valeurs *`region`* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la Référence générale d'Amazon Web Services.

> **Note**
>
> Pour Debian Server 8, seule l'architecture x86_64 est prise en charge.

x86_64

```
wget https://s3.region.amazonaws.com/amazon-ssm-region/latest/debian_amd64/amazon-ssm-agent.deb
```

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

Consultez l'exemple suivant.

```
wget https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/debian_amd64/amazon-ssm-agent.deb
```

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

ARM64

```
wget https://s3.region.amazonaws.com/amazon-ssm-region/latest/debian_arm64/amazon-ssm-agent.deb
```

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

Consultez l'exemple suivant.

```
wget https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/debian_arm64/
amazon-ssm-agent.deb
```

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

Installation manuelle de SSM Agent sur les instances Oracle Linux

Les Amazon Machine Images (AMIs) pour Oracle Linux cela sont fournis par AWS do not come with AWS Systems Manager Agent (SSM Agent) préinstallé par défaut. Pour obtenir une liste des AMIs gérées par AWS sur lesquelles l'agent peut être préinstallé, consultez la rubrique Rechercher AMIs avec le SSM Agent préinstallé.

Utilisez les informations de cette section pour vous aider à installer ou à réinstaller manuellement SSM Agent sur une instance Oracle Linux.

Avant de commencer

Avant d'installer SSM Agent sur une instance Oracle Linux, notez ce qui suit :

- Pour obtenir des informations importantes qui s'appliquent à l'installation de SSM Agent sur tous les systèmes d'exploitation Linux, consultez la rubrique Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Linux.
- Si vous utilisez une commande `yum` pour mettre à jour l'SSM Agent sur un nœud géré après que l'agent a été installé ou mis à jour en utilisant le document SSM `AWS-UpdateSSMAgent`, le message suivant peut s'afficher : « Warning: RPMDB altered outside of yum (Avertissement : RPMDB modifié en dehors de yum) ». Ce message est prévu pour s'afficher. Il peut être ignoré sans risque.

Rubriques

- Commandes d'installation rapide pour SSM Agent sur Oracle Linux
- Création de commandes d'installation d'agent personnalisées pour Oracle Linux dans votre région

Commandes d'installation rapide pour SSM Agent sur Oracle Linux

Suivez ces étapes pour installer manuellement SSM Agent sur une seule instance. Cette procédure utilise les fichiers d'installation disponibles dans le monde entier.

Pour installer SSM Agent sur Oracle Linux à l'aide de commandes de copier-coller rapides

1. Connectez-vous à l'instance Oracle Linux en utilisant votre méthode préférée, telle que SSH.
2. Copiez la commande suivante et exécutez-la sur l'instance.

   > **Note**
   >
   > Même si l'URL de la commande suivante inclus un répertoire `ec2-downloads-windows`, il s'agit des fichiers d'installation globaux corrects pour Oracle Linux.

   x86_64

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

3. (Recommandé) Exécutez la commande suivante pour vérifier que l'agent est en cours d'exécution.

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

   Dans la plupart des cas, la commande signale que l'agent est en cours d'exécution, comme illustré dans l'exemple suivant.

   ```
   amazon-ssm-agent.service - amazon-ssm-agent
   Loaded: loaded (/usr/lib/systemd/system/amazon-ssm-agent.service; enabled; vendor
    preset: enabled)
   Active: active (running) since Wed 2021-10-20 19:09:29 UTC; 4min 6s ago
              --truncated--
   ```

   Dans de rares cas, la commande signale que l'agent est installé, mais qu'il n'est pas en cours d'exécution, comme illustré dans l'exemple suivant.

   ```
   amazon-ssm-agent.service - amazon-ssm-agent
   ```

```
Loaded: loaded (/usr/lib/systemd/system/amazon-ssm-agent.service; enabled; vendor
 preset: enabled)
Active: inactive (dead) since Wed 2021-10-20 22:16:41 UTC; 18s ago
            --truncated--
```

Pour activer l'agent dans ces cas-là, exécutez les commandes suivantes.

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

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

Création de commandes d'installation d'agent personnalisées pour Oracle Linux dans votre région

Lorsque vous installez SSM Agent sur plusieurs instances utilisant un script ou un modèle, nous vous recommandons d'utiliser les fichiers d'installation stockés dans l' Région AWS dans laquelle vous travaillez.

S'il s'agit des commandes suivantes, nous fournissons des exemples qui utilisent un compartiment S3 accessible au public dans la région USA Est (Ohio) (`us-east-2`).

> **Tip**
>
> Vous pouvez également remplacer une URL globale dans la procédure Commandes d'installation rapide pour SSM Agent sur Oracle Linux plus avant dans cette rubrique, avec une URL régionale personnalisée que vous créez.

Dans la commande ci-après, remplacez *`region`* (région) par vos propres informations. Pour obtenir une liste des valeurs *`region`* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la Référence générale d'Amazon Web Services.

x86_64

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

Consultez l'exemple suivant.

```
sudo yum install -y https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/
linux_amd64/amazon-ssm-agent.rpm
```

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

Les Amazon Machine Images (AMIs) pour Red Hat Enterprise Linux (RHEL) fournis par ne sont AWS pas fournis avec AWS Systems Manager Agent (SSM Agent) préinstallé par défaut. Pour obtenir la liste des AWS sites gérés AMIs sur lesquels l'agent peut être préinstallé, consultez Rechercher AMIs avec le SSM Agent préinstallé.

Utilisez les informations de cette section pour vous aider à installer ou à réinstaller manuellement SSM Agent sur une instance RHEL.

Avant de commencer

Avant d'installer SSM Agent sur une instance RHEL, notez ce qui suit :

- Pour obtenir des informations importantes qui s'appliquent à l'installation de SSM Agent sur tous les systèmes d'exploitation Linux, consultez la rubrique Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Linux.
- Si vous utilisez une commande `yum` pour mettre à jour l'SSM Agent sur un nœud géré après que l'agent a été installé ou mis à jour en utilisant le document SSM `AWS-UpdateSSMAgent`, le message suivant peut s'afficher : « Warning: RPMDB altered outside of yum (Avertissement : RPMDB modifié en dehors de yum) ». Ce message est prévu pour s'afficher. Il peut être ignoré sans risque.

Rubriques

- Installation de SSM Agent sur RHEL 8.x et 9.x
- Installation de SSM Agent sur RHEL 7.x
- Installation de SSM Agent sur RHEL 6.x

Installation de SSM Agent sur RHEL 8.x et 9.x

Les Amazon Machine Images (AMIs) pour RHEL 8 et 9 fournis par ne sont AWS pas fournis avec AWS Systems Manager Agent (SSM Agent) préinstallé par défaut. Utilisez les informations de cette page pour vous aider à installer ou à réinstaller l'agent sur les instances RHEL 8 et 9.

Avant de commencer

Avant d'installer SSM Agent sur une instance RHEL 8 ou 9, notez ce qui suit :

- Assurez-vous que Python 2 ou Python 3 est installé sur votre instance RHEL 8 ou 9. Ceci est nécessaire pour que SSM Agent fonctionne correctement.

Rubriques

Commandes d'installation rapide pour SSM Agent sur RHEL 8 ou 9

Suivez ces étapes pour installer manuellement SSM Agent sur une seule instance. Cette procédure utilise les fichiers d'installation disponibles dans le monde entier.

Pour installer SSM Agent sur RHEL 8.x ou 9.x

1. Connectez-vous à votre instance RHEL 8 ou 9 en utilisant votre méthode préférée, telle que SSH.
2. Copiez la commande de l'architecture de votre instance et exécutez-la sur l'instance.

> **Note**
>
> Même si les URL des commandes suivantes incluent un répertoire `ec2-downloads-windows`, il s'agit des fichiers d'installation globaux corrects pour RHEL 8 et 9.

Instances x86_64

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

Instances ARM64

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

3. (Recommandé) Exécutez la commande suivante pour vérifier que l'agent est en cours d'exécution.

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

Dans la plupart des cas, la commande signale que l'agent est en cours d'exécution, comme illustré dans l'exemple suivant.

```
# amazon-ssm-agent.service - amazon-ssm-agent
   Loaded: loaded (/etc/systemd/system/amazon-ssm-agent.service; enabled; vendo>
   Active: active (running) since Tue 2022-04-19 16:40:41 UTC; 9s ago
 Main PID: 4898 (amazon-ssm-agen)
    Tasks: 14 (limit: 4821)
   Memory: 34.6M
   CGroup: /system.slice/amazon-ssm-agent.service
           ##4898 /usr/bin/amazon-ssm-agent
           ##4954 /usr/bin/ssm-agent-worker
            --truncated--
```

Dans de rares cas, la commande signale que l'agent est installé, mais qu'il n'est pas en cours d'exécution, comme illustré dans l'exemple suivant.

```
# amazon-ssm-agent.service - amazon-ssm-agent
   Loaded: loaded (/etc/systemd/system/amazon-ssm-agent.service; enabled; vendo>
   Active: inactive (dead) since Tue 2022-04-19 16:42:05 UTC; 2s ago
            --truncated--
```

Pour activer l'agent dans ces cas-là, exécutez les commandes suivantes.

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

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

Création de commandes d'installation d'agent personnalisées pour RHEL 8 et 9 dans votre région

Lorsque vous installez SSM Agent sur plusieurs instances utilisant un script ou un modèle, nous vous recommandons d'utiliser les fichiers d'installation stockés dans l' Région AWS dans laquelle vous travaillez.

S'il s'agit des commandes suivantes, nous fournissons des exemples qui utilisent un compartiment S3 accessible au public dans la région USA Est (Ohio) (`us-east-2`).

> **Tip**
>
> Vous pouvez également remplacer une URL globale dans la procédure Commandes d'installation rapide pour SSM Agent sur RHEL 8 ou 9 plus avant dans cette rubrique, avec une URL régionale personnalisée que vous créez.

Dans la commande ci-après, remplacez *`region`* (région) par vos propres informations. Pour obtenir une liste des valeurs *`region`* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la Référence générale d'Amazon Web Services.

x86_64

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

Consultez l'exemple suivant.

```
sudo dnf install -y https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/
linux_amd64/amazon-ssm-agent.rpm
```

ARM64

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

Consultez l'exemple suivant.

```
sudo dnf install -y https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/
linux_arm64/amazon-ssm-agent.rpm
```

Installation de SSM Agent sur RHEL 7.x

Les Amazon Machine Images (AMIs) pour RHEL 7 fournies par AWS n'incluent pas l'agent AWS Systems Manager (SSM Agent) par défaut. Utilisez les informations de cette page pour vous aider à installer ou à réinstaller l'agent sur les instances RHEL 7.

Rubriques

Commandes d'installation rapide pour SSM Agent sur RHEL 7

Suivez ces étapes pour installer manuellement SSM Agent sur une seule instance. Cette procédure utilise les fichiers d'installation disponibles dans le monde entier.

Installer l'SSM Agent sur RHEL 7.x

1. Connectez-vous à l'instance RHEL 7 en utilisant votre méthode préférée, telle que SSH.
2. Copiez la commande de l'architecture de votre instance et exécutez-la sur l'instance.

   > **Note**
   >
   > Même si les URL des commandes suivantes incluent un répertoire `ec2-downloads-windows`, il s'agit des fichiers d'installation globaux corrects pour RHEL 7.

   Instances x86_64

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

   Instances ARM64

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

3. (Recommandé) Exécutez la commande suivante pour vérifier que l'agent est en cours d'exécution.

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

   Dans la plupart des cas, la commande signale que l'agent est en cours d'exécution, comme illustré dans l'exemple suivant.

```
# amazon-ssm-agent.service - amazon-ssm-agent
   Loaded: loaded (/etc/systemd/system/amazon-ssm-agent.service; enabled; vendor
 preset: disabled)
   Active: active (running) since Tue 2022-04-19 16:47:36 UTC; 22s ago
 Main PID: 1342 (amazon-ssm-agen)
   CGroup: /system.slice/amazon-ssm-agent.service
           ##1342 /usr/bin/amazon-ssm-agent
           ##1362 /usr/bin/ssm-agent-worker
            --truncated--
```

Dans de rares cas, la commande signale que l'agent est installé, mais qu'il n'est pas en cours d'exécution, comme illustré dans l'exemple suivant.

```
# amazon-ssm-agent.service - amazon-ssm-agent
   Loaded: loaded (/etc/systemd/system/amazon-ssm-agent.service; enabled; vendor
 preset: disabled)
   Active: inactive (dead) since Tue 2022-04-19 16:48:56 UTC; 5s ago
  Process: 1342 ExecStart=/usr/bin/amazon-ssm-agent (code=exited, status=0/SUCCESS)
 Main PID: 1342 (code=exited, status=0/SUCCESS)
            --truncated--
```

Pour activer l'agent dans ces cas-là, exécutez les commandes suivantes.

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

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

Création de commandes d'installation d'agent personnalisées pour RHEL 7 dans votre région

Lorsque vous installez SSM Agent sur plusieurs instances utilisant un script ou un modèle, nous vous recommandons d'utiliser les fichiers d'installation stockés dans l'Région AWS dans laquelle vous travaillez.

S'il s'agit des commandes suivantes, nous fournissons des exemples qui utilisent un compartiment S3 accessible au public dans la région USA Est (Ohio) (`us-east-2`).

> **Tip**
>
> Vous pouvez également remplacer une URL globale dans la procédure Commandes d'installation rapide pour SSM Agent sur RHEL 7 plus avant dans cette rubrique, avec une URL régionale personnalisée que vous créez.

Dans la commande ci-après, remplacez *region* (région) par vos propres informations. Pour obtenir une liste des valeurs *region* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la Référence générale d'Amazon Web Services.

x86_64

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

Consultez l'exemple suivant.

```
sudo yum install -y https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/
linux_amd64/amazon-ssm-agent.rpm
```

ARM64

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

Consultez l'exemple suivant.

```
sudo yum install -y https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/
linux_arm64/amazon-ssm-agent.rpm
```

Installation de SSM Agent sur RHEL 6.x

Les Amazon Machine Images (AMIs) pour RHEL 6 fournies par AWS n'incluent pas l'agent AWS Systems Manager (SSM Agent) par défaut. Utilisez les informations de cette page pour vous aider à installer ou à réinstaller l'agent sur les instances RHEL 6.

Rubriques

- Commandes d'installation rapide pour SSM Agent sur RHEL 6

- Création de commandes d'installation d'agent personnalisées pour RHEL 6 dans votre région

Commandes d'installation rapide pour SSM Agent sur RHEL 6

Suivez ces étapes pour installer manuellement SSM Agent sur une seule instance. Cette procédure utilise les fichiers d'installation disponibles dans le monde entier.

Installer l'SSM Agent sur RHEL 6.x

1. Connectez-vous à l'instance RHEL 6 en utilisant votre méthode préférée, telle que SSH.
2. Copiez la commande de l'architecture de votre instance et exécutez-la sur l'instance.

> **Note**
>
> Même si les URL des commandes suivantes incluent un répertoire `ec2-downloads-windows`, il s'agit des fichiers d'installation globaux corrects pour RHEL 6.
> Les commandes suivantes spécifient le répertoire des versions `3.0.1479.0` au lieu du répertoire `latest`. C'est parce que les versions 3.1 et ultérieures de SSM Agent ne sont pas prises en charge pour RHEL 6.

Instances x86_64

```
sudo yum install -y https://s3.amazonaws.com/ec2-downloads-windows/
SSMAgent/3.0.1479.0/linux_amd64/amazon-ssm-agent.rpm
```

Instances x86

```
sudo yum install -y https://s3.amazonaws.com/ec2-downloads-windows/
SSMAgent/3.0.1479.0/linux_386/amazon-ssm-agent.rpm
```

3. (Recommandé) Exécutez la commande suivante pour vérifier que l'agent est en cours d'exécution.

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

Dans la plupart des cas, la commande signale que l'agent est en cours d'exécution, comme illustré dans l'exemple suivant.

```
amazon-ssm-agent start/running, process 1788
```

Dans de rares cas, la commande signale que l'agent est installé, mais qu'il n'est pas en cours d'exécution, comme illustré dans l'exemple suivant.

```
amazon-ssm-agent stop/waiting
```

Dans ces cas, exécutez la commande suivante pour activer l'agent.

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

Création de commandes d'installation d'agent personnalisées pour RHEL 6 dans votre région

Lorsque vous installez SSM Agent sur plusieurs instances utilisant un script ou un modèle, nous vous recommandons d'utiliser les fichiers d'installation stockés dans l'Région AWS dans laquelle vous travaillez.

S'il s'agit des commandes suivantes, nous fournissons des exemples qui utilisent un compartiment S3 accessible au public dans la région USA Est (Ohio) (`us-east-2`).

> **Tip**
>
> Vous pouvez également remplacer une URL globale dans la procédure Commandes d'installation rapide pour SSM Agent sur RHEL 6 plus avant dans cette rubrique, avec une URL régionale personnalisée que vous créez.

Dans la commande ci-après, remplacez *`region`* (région) par vos propres informations. Pour obtenir une liste des valeurs *`region`* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la Référence générale d'Amazon Web Services.

> **Note**
>
> Les commandes suivantes spécifient le répertoire des versions `3.0.1390.0` au lieu du répertoire `latest`. C'est parce que les versions 3.1 et ultérieures de SSM Agent ne sont pas prises en charge pour RHEL 6.

x86_64

```
sudo yum install -y https://s3.region.amazonaws.com/amazon-ssm-region/3.0.1479.0/
linux_amd64/amazon-ssm-agent.rpm
```

Consultez l'exemple suivant.

```
sudo yum install -y https://s3.us-east-2.amazonaws.com/amazon-ssm-us-
east-2/3.0.1479.0/linux_amd64/amazon-ssm-agent.rpm
```

x86

```
sudo yum install -y https://s3.region.amazonaws.com/amazon-ssm-region/3.0.1479.0/
linux_386/amazon-ssm-agent.rpm
```

Consultez l'exemple suivant.

```
sudo yum install -y https://s3.us-east-2.amazonaws.com/amazon-ssm-us-
east-2/3.0.1479.0/linux_386/amazon-ssm-agent.rpm
```

Installation manuelle de SSM Agent sur les instances Rocky Linux

Les Amazon Machine Images (AMIs) pour Rocky Linux cela sont fournis par AWS do not come with AWS Systems Manager Agent (SSM Agent) préinstallé par défaut. Pour obtenir une liste des AMIs gérées par AWS sur lesquelles l'agent peut être préinstallé, consultez la rubrique Rechercher AMIs avec le SSM Agent préinstallé.

Utilisez les informations de cette section pour vous aider à installer ou à réinstaller manuellement SSM Agent sur une instance Rocky Linux.

Avant de commencer

Avant d'installer SSM Agent sur une instance Rocky Linux, notez ce qui suit :

- Pour obtenir des informations importantes qui s'appliquent à l'installation de SSM Agent sur tous les systèmes d'exploitation Linux, consultez la rubrique Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Linux.

Rubriques

- Commandes d'installation rapide pour SSM Agent sur Rocky Linux
- Création de commandes d'installation d'agent personnalisées pour Rocky Linux dans votre région

Commandes d'installation rapide pour SSM Agent sur Rocky Linux

Suivez ces étapes pour installer manuellement SSM Agent sur une seule instance. Cette procédure utilise les fichiers d'installation disponibles dans le monde entier.

Avant de commencer

Avant d'installer SSM Agent sur une instance Rocky Linux, notez ce qui suit :

- Assurez-vous que Python 2 ou Python 3 est installé sur votre instance Rocky Linux. Ceci est nécessaire pour que SSM Agent fonctionne correctement.

Pour installer l'SSM Agent sur Rocky Linux

1. Connectez-vous à l'instance Rocky Linux en utilisant votre méthode préférée, telle que SSH.
2. Copiez la commande de l'architecture de votre instance et exécutez-la sur l'instance.

   > **Note**
   >
   > Même si les URL des commandes suivantes incluent un répertoire `ec2-downloads-windows`, il s'agit des fichiers d'installation globaux corrects pour Rocky Linux.

   Instances x86_64

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

   Instances ARM64

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

3. (Recommandé) Exécutez la commande suivante pour vérifier que l'agent est en cours d'exécution.

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

Dans la plupart des cas, la commande signale que l'agent est en cours d'exécution, comme illustré dans l'exemple suivant.

```
# amazon-ssm-agent.service - amazon-ssm-agent
   Loaded: loaded (/etc/systemd/system/amazon-ssm-agent.service; enabled; vendo>
   Active: active (running) since Tue 2022-04-19 16:40:41 UTC; 9s ago
 Main PID: 4898 (amazon-ssm-agen)
    Tasks: 14 (limit: 4821)
   Memory: 34.6M
   CGroup: /system.slice/amazon-ssm-agent.service
           ##4898 /usr/bin/amazon-ssm-agent
           ##4954 /usr/bin/ssm-agent-worker
            --truncated--
```

Dans de rares cas, la commande signale que l'agent est installé, mais qu'il n'est pas en cours d'exécution, comme illustré dans l'exemple suivant.

```
# amazon-ssm-agent.service - amazon-ssm-agent
   Loaded: loaded (/etc/systemd/system/amazon-ssm-agent.service; enabled; vendo>
   Active: inactive (dead) since Tue 2022-04-19 16:42:05 UTC; 2s ago
            --truncated--
```

Pour activer l'agent dans ces cas-là, exécutez les commandes suivantes.

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

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

Création de commandes d'installation d'agent personnalisées pour Rocky Linux dans votre région

Lorsque vous installez SSM Agent sur plusieurs instances utilisant un script ou un modèle, nous vous recommandons d'utiliser les fichiers d'installation stockés dans l' Région AWS dans laquelle vous travaillez.

S'il s'agit des commandes suivantes, nous fournissons des exemples qui utilisent un compartiment S3 accessible au public dans la région USA Est (Ohio) (`us-east-2`).

> **Tip**
>
> Vous pouvez également remplacer une URL globale dans la procédure Commandes d'installation rapide pour SSM Agent sur Rocky Linux plus avant dans cette rubrique, avec une URL régionale personnalisée que vous créez.

Dans la commande ci-après, remplacez *region* (région) par vos propres informations. Pour obtenir une liste des valeurs *region* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la Référence générale d'Amazon Web Services.

x86_64

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

Consultez l'exemple suivant.

```
sudo dnf install -y https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/
linux_amd64/amazon-ssm-agent.rpm
```

ARM64

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

Consultez l'exemple suivant.

```
sudo dnf install -y https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/
linux_arm64/amazon-ssm-agent.rpm
```

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

Dans la plupart des cas, les Amazon Machine Images (AMIs) pour SUSE Linux Enterprise Server (SLES) fournis par sont AWS fournis avec AWS Systems Manager Agent (SSM Agent) préinstallé par défaut. Pour plus d'informations, consultez Rechercher AMIs avec le SSM Agent préinstallé.

Dans le cas où SSM Agent n'est pas préinstallé sur une nouvelle instance SLES, ou si vous devez réinstaller manuellement l'agent, utilisez les informations de cette page pour vous aider.

Avant de commencer

Avant d'installer SSM Agent sur une instance SLES, notez ce qui suit :

- Pour obtenir des informations importantes qui s'appliquent à l'installation de SSM Agent sur tous les systèmes d'exploitation Linux, consultez la rubrique Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Linux.

Rubriques

- Commandes d'installation rapide pour SSM Agent sur SLES
- Création de commandes d'installation d'agent personnalisées pour SLES dans votre région

Commandes d'installation rapide pour SSM Agent sur SLES

Suivez ces étapes pour installer manuellement SSM Agent sur une seule instance. Cette procédure utilise les fichiers d'installation disponibles dans le monde entier.

Pour installer SSM Agent sur SLES à l'aide de commandes de copier-coller rapides

1. Connectez-vous à l'instance SLES en utilisant votre méthode préférée, telle que SSH.
2. Option 1 : Utiliser une commande `zypper`

   - Exécutez la commande suivante :

   ```
   sudo zypper install amazon-ssm-agent
   ```

   - Saisissez y en réponse à l'invite.

   Option 2 : Utiliser une commande `rpm`

   - Créez un répertoire temporaire sur l'instance.

   ```
   mkdir /tmp/ssm
   ```

   - Passez au répertoire temporaire.

   ```
   cd /tmp/ssm
   ```

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

  > **Note**
  >
  > Même si les URL des commandes suivantes incluent un répertoire `ec2-downloads-windows`, il s'agit des fichiers d'installation globaux corrects pour SLES.

  Instances x86_64 :

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

  Instances ARM64 :

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

- Exécutez la commande suivante.

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

- (Recommandé) Exécutez la commande suivante pour vérifier que l'agent est en cours d'exécution.

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

  Dans la plupart des cas, la commande signale que l'agent est en cours d'exécution, comme illustré dans l'exemple suivant.

  ```
  # amazon-ssm-agent.service - amazon-ssm-agent
   Loaded: loaded (/usr/lib/systemd/system/amazon-ssm-agent.service; enabled;
   vendor preset: disabled)
   Active: active (running) since Mon 2022-02-21 23:13:28 UTC; 7s ago
   Main PID: 2102 (amazon-ssm-agen)
   Tasks: 15 (limit: 512)
   CGroup: /system.slice/amazon-ssm-agent.service
   ##2102 /usr/sbin/amazon-ssm-agent
  ```

```
 ##2107 /usr/sbin/ssm-agent-worker
           --truncated--
```

Dans de rares cas, la commande signale que l'agent est installé, mais qu'il n'est pas en cours d'exécution, comme illustré dans l'exemple suivant.

```
# amazon-ssm-agent.service - amazon-ssm-agent
   Loaded: loaded (/usr/lib/systemd/system/amazon-ssm-agent.service; disabled;
 vendor preset: disabled)
   Active: inactive (dead)
           --truncated--
```

Pour activer l'agent dans ces cas-là, exécutez les commandes suivantes.

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

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

Création de commandes d'installation d'agent personnalisées pour SLES dans votre région

Lorsque vous installez SSM Agent sur plusieurs instances utilisant un script ou un modèle, nous vous recommandons d'utiliser les fichiers d'installation stockés dans l' Région AWS dans laquelle vous travaillez.

S'il s'agit des commandes suivantes, nous fournissons des exemples qui utilisent un compartiment S3 accessible au public dans la région USA Est (Ohio) (`us-east-2`).

> **Tip**
>
> Vous pouvez également remplacer une URL globale dans la procédure Commandes d'installation rapides pour SSM Agent Amazon Linux 1 plus avant dans cette rubrique, avec une URL régionale personnalisée que vous créez.

Dans la commande ci-après, remplacez *`region`* (région) par vos propres informations. Pour obtenir une liste des valeurs *`region`* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la Référence générale d'Amazon Web Services.

x86_64

```
wget https://s3.region.amazonaws.com/amazon-ssm-region/latest/linux_amd64/amazon-ssm-agent.rpm
```

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

Consultez l'exemple suivant.

```
wget https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/linux_amd64/amazon-ssm-agent.rpm
```

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

ARM64

```
wget https://s3.region.amazonaws.com/amazon-ssm-region/latest/linux_arm64/amazon-ssm-agent.rpm
```

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

Consultez l'exemple suivant.

```
wget https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/linux_arm64/amazon-ssm-agent.rpm
```

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

Installation manuelle de SSM Agent sur les instances Ubuntu Server

Important

Avant l'installation de SSM Agent sur une version 64 bits de Ubuntu Server, assurez-vous d'utiliser les bons outils d'installation. À partir des Amazon Machine Images (AMI) portant la date 20180627, SSM Agent est préinstallé sur la version 16.04 à l'aide de packages Snap. Sur les instances créées à partir d'AMI antérieures, SSM Agent doit être installé à l'aide des

packages d'installation deb. Pour plus d'informations, consultez Choix de la version de SSM Agent à installer sur les instances Ubuntu Server 16.04 64 bits.

Dans la plupart des cas, les Amazon Machine Images Ubuntu Server (AMIs) sont fournis par AWS come with AWS Systems Manager Agent (SSM Agent) préinstallé par défaut. Pour plus d'informations, consultez Rechercher AMIs avec le SSM Agent préinstallé.

Dans le cas où SSM Agent n'est pas préinstallé sur une nouvelle instance Ubuntu Server, ou si vous devez réinstaller manuellement l'agent, utilisez les informations de cette section pour vous aider.

Avant de commencer

Avant d'installer SSM Agent sur une instance Ubuntu Server, notez ce qui suit :

- Pour obtenir des informations importantes qui s'appliquent à l'installation de SSM Agent sur tous les systèmes d'exploitation Linux, consultez la rubrique Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Linux.

Rubriques

Installation de SSM Agent sur Ubuntu Server 22.04 LTS, 20.10 STR et 20.04, 18.04 et 16.04 LTS 64 bits (Snap)

Avant de commencer

Avant l'installation de SSM Agent sur un Ubuntu Server 22.04 LTS, 20.10 STR et 20.04, 18.04 et 16.04 LTS 64 bits (Snap), notez les points suivants :

Installation de la version 16.04 par Snaps ou les installateurs deb

Sur Ubuntu Server 16.04, l'SSM Agent est installé à l'aide de Snaps ou de packages d'installation deb, en fonction de la version de l'AMI 16.04.

Emplacements des fichiers d'installation SSM Agent

Sur Ubuntu Server 22.04 LTS, 20.10 STR et 20.04, 18.04 et 16.04 LTS (avec Snap), les fichiers du programme d'installation de SSM Agent, 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 répertoire `/snap` vers le répertoire `/etc/amazon/ssm/`. Les fichiers journaux et de bibliothèque n'ont pas changé (`/var/lib/amazon/ssm`, `/var/log/amazon/ssm`).

Utilisation du canal Snap `candidate`

Le canal candidat dans le magasin de Snaps contient la dernière version de l'SSM Agent (y compris l'intégralité des dernières corrections de bogues) ; pas le canal stable. Pour en savoir plus sur les différences entre le canal candidat et le canal stable, veuillez consulter Niveaux de risque à l'adresse https://snapcraft.io/docs/channels.

Si vous voulez suivre les informations de version de l'SSM Agent sur le canal candidat, exécutez la commande suivante sur vos instances 64 bits Ubuntu Server 20.10 STR et 20.04, 18.04 et 16.04 LTS.

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

Snaps recommandés sur les versions 18.04 et ultérieures

Sur Ubuntu Server 22.04 LTS, 20.10 STR et 20.04 et 18.04 LTS, nous vous recommandons d'utiliser uniquement les Snaps. Vérifiez également qu'une seule instance de l'agent est installée et en cours d'exécution sur vos instances. Si vous voulez utiliser l'SSM Agent sans les Snaps, désinstallez l'SSM Agent. Ensuite, installez SSM Agent sous forme de paquet Debian en utilisant les instructions d'installation SSM Agent sur Ubuntu Server 16.04 et 14.04 64 bits (deb). Avant de procéder à l'installation, assurez-vous que vous n'avez pas de Snaps installé qui chevauche la liste des paquets que vous voulez gérer comme des paquets debian.

Message d'erreur `Maximum timeout exceeded`

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
```

Pour installer SSM Agent sur des instances Ubuntu Server 22.04 LTS, 20.10 STR et 20.04, 18.04 et 16.04 LTS 64 bits (avec un package Snap)

1. Par défaut, SSM Agent est installé sur des AMIs Ubuntu Server 22.04 LTS, 20.04, 18.04 et 16.04 LTS 64 bits avec un identifiant `20180627` ou ultérieur.

   Vous pouvez utiliser le script suivant si vous avez besoin d'installer l'SSM Agent 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
   ```

2. Exécutez la commande suivante afin de déterminer si l'SSM Agent 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
   ```

Installation de SSM Agent sur Ubuntu Server 16.04 et 14.04 64 bits (deb)

> **Important**
>
> Avant l'installation de SSM Agent sur une version 64 bits de Ubuntu Server, assurez-vous d'utiliser les bons outils d'installation. À partir des Amazon Machine Images (AMI) portant la date 20180627, SSM Agent est préinstallé sur la version 16.04 à l'aide de packages Snap. Sur les instances créées à partir d'AMI antérieures, SSM Agent doit être installé à l'aide des packages d'installation deb. Pour plus d'informations, consultez Choix de la version de SSM Agent à installer sur les instances Ubuntu Server 16.04 64 bits.Si SSM Agent est installé sur votre instance en conjonction avec un Snap et que vous installez ou mettez à jour SSM Agent en utilisant un paquet d'installation deb, l'installation ou les opérations de SSM Agent peuvent échouer.

Dans la plupart des cas, les Amazon Machine Images (AMIs) Ubuntu Server 16.04 fournis par sont AWS fournis avec AWS Systems Manager Agent (SSM Agent) préinstallé par défaut. Pour plus d'informations, consultez Rechercher AMIs avec le SSM Agent préinstallé.

Dans le cas où SSM Agent n'est pas préinstallé sur une nouvelle instance Ubuntu Server 16.04 antérieure à la version 20180627, que vous effectuez une installation sur Ubuntu Server 14.04, ou si vous devez réinstaller manuellement l'agent, utilisez les informations de cette page pour vous aider.

Commandes d'installation rapide pour SSM Agent sur Ubuntu Server 16.04 et 14.04 64 bits (deb)

Suivez ces étapes pour installer manuellement SSM Agent sur une seule instance. Cette procédure utilise les fichiers d'installation disponibles dans le monde entier.

Pour installer SSM Agent sur Ubuntu Server 16.04 et 14.04 64 bits (deb) à l'aide des commandes de copier-coller rapides

1. Connectez-vous à l'instance Ubuntu Server en utilisant votre méthode préférée, telle que SSH.
2. Exécutez la commande suivante pour créer un répertoire temporaire sur l'instance.

```
mkdir /tmp/ssm
```

3. Passez au répertoire temporaire.

```
cd /tmp/ssm
```

4. Exécutez les commandes suivantes.

> **Note**
>
> Même si les URL des commandes suivantes incluent un répertoire `ec2-downloads-windows`, il s'agit des fichiers d'installation globaux corrects pour Ubuntu Server 16.04 et 14.04 64 bits.

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

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

5. (Recommandé) Exécutez l'une des commandes suivantes afin de déterminer si l'SSM Agent est en cours d'exécution.

   Ubuntu Server 16.04

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

   Ubuntu Server14,04

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

   Dans la plupart des cas, la commande signale que l'agent est en cours d'exécution.

   Dans de rares cas, la commande signale que l'agent est installé, mais qu'il n'est pas en cours d'exécution, comme illustré dans l'exemple suivant.

6. 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
```

Création de commandes d'installation personnalisées pour SSM Agent sur Ubuntu Server 16.04 et 14.04 64 bits (deb) dans votre région

Lorsque vous installez SSM Agent sur plusieurs instances utilisant un script ou un modèle, nous vous recommandons d'utiliser les fichiers d'installation stockés dans l' Région AWS dans laquelle vous travaillez.

S'il s'agit des commandes suivantes, nous fournissons des exemples qui utilisent un compartiment S3 accessible au public dans la région USA Est (Ohio) (`us-east-2`).

> **Tip**
>
> Vous pouvez également remplacer une URL globale dans la procédure Commandes d'installation rapide pour SSM Agent sur Ubuntu Server 16.04 et 14.04 64 bits (deb) plus avant dans cette rubrique, avec une URL régionale personnalisée que vous créez.

Dans la commande ci-après, remplacez *`region`* (région) par vos propres informations. Pour obtenir une liste des valeurs *`region`* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la 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
```

Consultez l'exemple suivant.

```
wget https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/debian_amd64/amazon-ssm-agent.deb
```

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

Installation de SSM Agent sur Ubuntu Server 16.04 et 14.04 32 bits

Dans la plupart des cas, les Amazon Machine Images (AMIs) Ubuntu Server 16.04 fournis par sont AWS fournis avec AWS Systems Manager Agent (SSM Agent) préinstallé par défaut. Pour plus d'informations, consultez Rechercher AMIs avec le SSM Agent préinstallé.

Dans le cas où SSM Agent n'est pas préinstallé sur une nouvelle instance Ubuntu Server 16.04, que vous installez sur Ubuntu Server 14.04, ou si vous devez réinstaller manuellement l'agent, utilisez les informations de cette page pour vous aider.

Commandes d'installation rapide pour SSM Agent sur Ubuntu Server 16.04 et 14.04 32 bits (deb)

Suivez ces étapes pour installer manuellement SSM Agent sur une seule instance. Cette procédure utilise les fichiers d'installation disponibles dans le monde entier.

Pour installer SSM Agent sur Ubuntu Server 16.04 et 14.04 32 bits (deb) à l'aide des commandes de copier-coller rapides

1. Connectez-vous à l'instance Ubuntu Server en utilisant votre méthode préférée, telle que SSH.
2. Exécutez la commande suivante pour créer un répertoire temporaire sur l'instance.

```
mkdir /tmp/ssm
```

3. Passez au répertoire temporaire.

```
cd /tmp/ssm
```

4. Exécutez les commandes suivantes.

> **Note**
>
> Même si les URL des commandes suivantes incluent un répertoire `ec2-downloads-windows`, il s'agit des fichiers d'installation globaux corrects pour Ubuntu Server 16.04 et 14.04 32 bits.

```
wget https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/debian_386/
amazon-ssm-agent.deb
```

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

5. (Recommandé) Exécutez l'une des commandes suivantes afin de déterminer si l'SSM Agent est en cours d'exécution.

   Ubuntu Server 16.04

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

   Ubuntu Server14,04

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

   Dans la plupart des cas, la commande signale que l'agent est en cours d'exécution.

   Dans de rares cas, la commande signale que l'agent est installé, mais qu'il n'est pas en cours d'exécution, comme illustré dans l'exemple suivant.

6. 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
   ```

Création de commandes d'installation personnalisées pour SSM Agent sur Ubuntu Server 16.04 et 14.04 32 bits (deb) dans votre région

Lorsque vous installez SSM Agent sur plusieurs instances utilisant un script ou un modèle, nous vous recommandons d'utiliser les fichiers d'installation stockés dans l' Région AWS dans laquelle vous travaillez.

S'il s'agit des commandes suivantes, nous fournissons des exemples qui utilisent un compartiment S3 accessible au public dans la région USA Est (Ohio) (`us-east-2`).

> **Tip**
>
> Vous pouvez également remplacer une URL globale dans la procédure Commandes d'installation rapide pour SSM Agent sur Ubuntu Server 16.04 et 14.04 32 bits (deb) plus avant dans cette rubrique, avec une URL régionale personnalisée que vous créez.

Dans la commande ci-après, remplacez *region* (région) par vos propres informations. Pour obtenir une liste des valeurs *region* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la 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
```

Consultez l'exemple suivant.

```
wget https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/debian_386/amazon-ssm-agent.deb
```

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

Choix de la version de SSM Agent à installer sur les instances Ubuntu Server 16.04 64 bits

> **Important**
>
> Avant l'installation de SSM Agent sur une version 64 bits de Ubuntu Server, assurez-vous d'utiliser les bons outils d'installation. À partir des Amazon Machine Images (AMI) portant la date 20180627, SSM Agent est préinstallé sur la version 16.04 à l'aide de packages Snap. Sur les instances créées à partir d'AMI antérieures, SSM Agent doit être installé à l'aide des packages d'installation deb. Pour de plus amples informations, veuillez consulter Choix de la version de SSM Agent à installer sur les instances Ubuntu Server 16.04 64 bits
> Sachez que si une instance contient plusieurs installations de l'SSM Agent (par exemple, une effectuée à l'aide d'un snap et l'autre à l'aide d'un programme d'installation deb), vos opérations d'agent ne fonctionneront pas correctement.

Vous pouvez vérifier la date de création de l'ID de l'AMI source d'une instance selon l'une des méthodes suivantes. Ces procédures s'appliquent uniquement aux AMIs gérées par AWS.

Vérification de la date de création de l'ID d'une AMI source (console)

1. Ouvrez la console Amazon EC2 sur https://console.aws.amazon.com/ec2/.
2. Dans le panneau de navigation de gauche, sélectionnez instances.
3. Sélectionnez une instance.
4. Sous l'onglet Details (Détails), vérifiez la présence d'un identifiant YYYYMMDD dans la valeur du champ AMI name (Nom de l'). Par exemple : `ubuntu/images/hvm-ssd/ubuntu-xenial-16.04-amd64-server-20180627`.

Vérification de la date de création de l'ID d'une AMI source (AWS CLI)

- Exécutez la commande suivante.

```
aws ec2 describe-images --image-ids ami-id
```

ami-id désigne l'ID d'une AMI fournie par AWS. `ami-07c8bc5c1ce9598c3` par exemple.

En cas de réussite, la commande renvoie des informations semblables aux suivantes, dans lesquelles vous pouvez vérifier les champs `CreationDate` et `Name` afin d'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"
        }
    ]
```

```
}
```

Configuration SSM Agent pour utiliser un proxy sur les nœuds Linux

Vous pouvez configurer AWS Systems Manager Agent (SSM Agent) pour qu'il communique par le biais d'un proxy HTTP en créant un fichier de configuration de remplacement et en ajoutant `http_proxy` des `no_proxy` paramètres au fichier. `https_proxy` Un fichier de remplacement conserve également les paramètres de proxy si vous installez des versions plus récentes ou antérieures de l'SSM Agent. Cette section comprend les procédures de création d'un fichier de remplacement des environnements upstart et systemd. Si vous avez l'intention de l'utiliserSession Manager, notez que les serveurs proxy HTTPS ne sont pas pris en charge.

Rubriques

- Configurer l'SSM Agent afin d'utiliser un proxy (Upstart)
- Configuration de l'SSM Agent afin d'utiliser un proxy (systemd)

Configurer l'SSM Agent afin d'utiliser un proxy (Upstart)

Procédez comme suit pour créer un fichier de configuration de remplacement pour un environnement `upstart`.

Pour configurer l'SSM Agent afin d'utiliser un proxy (Upstart)

1. Connectez-vous à l'instance gérée sur laquelle vous avez installé l'SSM Agent.
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=IP address for instance metadata services (IMDS)
   ```

   Pour un serveur proxy HTTPS :

   ```
   env http_proxy=http://hostname:port
   env https_proxy=https://hostname:port
   ```

```
env no_proxy=IP address for instance metadata services (IMDS)
```

> **Important**
>
> Ajoutez le `no_proxy` paramètre au fichier et spécifiez l'adresse IP. L'adresse IP de `no_proxy` est le point de terminaison des services de métadonnées d'instance (IMDS) pour Systems Manager. Si vous ne le spécifiez pas`no_proxy`, les appels à Systems Manager prennent l'identité du service proxy (si la solution de secours IMDSv1 est activée) ou les appels à Systems Manager échouent (si IMDSv2 est appliqué).
>
> - Pour IPv4, spécifiez`no_proxy=169.254.169.254`.
> - Pour IPv6, spécifiez`no_proxy=[fd00:ec2::254]`. L'adresse IPv6 du service de métadonnées d'instance est compatible avec les commandes IMDSv2. L'adresse IPv6 n'est accessible que sur les instances créées sur le système AWS Nitro. Pour plus d'informations, consultez Comment fonctionne le service de métadonnées d'instance version 2 dans le guide de l'utilisateur Amazon EC2.

3. Enregistrez le fichier sous le nom `amazon-ssm-agent.override` à l'emplacement suivant : `/etc/init/`
4. Arrêtez et redémarrez l'SSM Agent à l'aide des commandes suivantes.

```
sudo service stop amazon-ssm-agent
sudo service start amazon-ssm-agent
```

> **Note**
>
> Pour de plus amples informations sur l'utilisation de fichiers `.override` dans les environnements Upstart, veuillez consulter init: Upstart init daemon job configuration (init : configuration de tâche Upstart init daemon).

Configuration de l'SSM Agent afin d'utiliser un proxy (systemd)

Procédez comme suit pour configurer SSM Agent de sorte à utiliser un proxy dans un environnement `systemd`.

> **Note**
>
> Certaines étapes de cette procédure contiennent des instructions explicites pour les instances Ubuntu Server pour lesquelles l'SSM Agent a été installé à l'aide de Snap.

1. Connectez-vous à l'instance sur laquelle vous avez installé l'SSM Agent.
2. Exécutez l'une des commandes suivantes, selon le type de système d'exploitation.

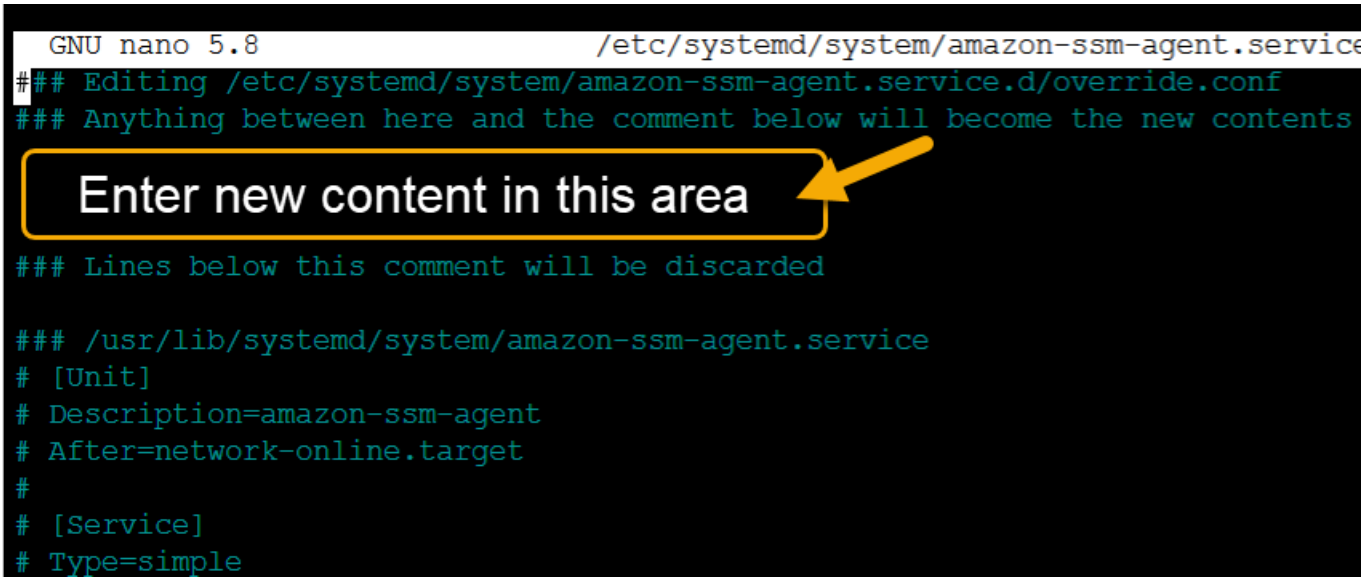
   - Sur les instances Ubuntu Server où l'SSM Agent est installé en utilisant un snap :

     ```
     sudo systemctl edit snap.amazon-ssm-agent.amazon-ssm-agent
     ```

     Sur d'autres systèmes d'exploitation

     ```
     sudo 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.

   Assurez-vous de saisir les informations au-dessus du commentaire qui dit `### Lines below this comment will be discarded` « », comme indiqué dans l'image suivante.



   Pour un serveur proxy HTTP :

   ```
   [Service]
   ```

```
Environment="http_proxy=http://hostname:port"
Environment="https_proxy=http://hostname:port"
Environment="no_proxy=IP address for instance metadata services (IMDS)"
```

Pour un serveur proxy HTTPS :

```
[Service]
Environment="http_proxy=http://hostname:port"
Environment="https_proxy=https://hostname:port"
Environment="no_proxy=IP address for instance metadata services (IMDS)"
```

> **Important**
>
> Ajoutez le `no_proxy` paramètre au fichier et spécifiez l'adresse IP. L'adresse IP de `no_proxy` est le point de terminaison des services de métadonnées d'instance (IMDS) pour Systems Manager. Si vous ne le spécifiez pas`no_proxy`, les appels à Systems Manager prennent l'identité du service proxy (si la solution de secours IMDSv1 est activée) ou les appels à Systems Manager échouent (si IMDSv2 est appliqué).
>
> - Pour IPv4, spécifiez`no_proxy=169.254.169.254`.
> - Pour IPv6, spécifiez`no_proxy=[fd00:ec2::254]`. L'adresse IPv6 du service de métadonnées d'instance est compatible avec les commandes IMDSv2. L'adresse IPv6 n'est accessible que sur les instances créées sur le système AWS Nitro. Pour plus d'informations, consultez Comment fonctionne le service de métadonnées d'instance version 2 dans le guide de l'utilisateur Amazon EC2.

4. Enregistrez vos modifications. Le système crée automatiquement l'un des fichiers suivants, en fonction du type de système d'exploitation.

   - Sur les instances Ubuntu Server où l'SSM Agent est installé en utilisant un snap :

     `/etc/systemd/system/snap.amazon-ssm-agent.amazon-ssm-agent.service.d/`**`override.conf`**

   - Sur les instances Amazon Linux 2 et Amazon Linux 2023 :

     `/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 l'SSM Agent en utilisant l'une des commandes suivantes, en fonction du type de système d'exploitation.

   - Sur les instances Ubuntu Server installées à l'aide de Snap :

     ```
     sudo systemctl daemon-reload && sudo systemctl restart snap.amazon-ssm-agent.amazon-ssm-agent
     ```

   - Sur d'autres systèmes d'exploitation

     ```
     sudo systemctl daemon-reload && sudo systemctl restart amazon-ssm-agent
     ```

> **Note**
>
> Pour de plus amples informations sur l'utilisation des fichiers `.override` dans les environnements systemd, veuillez consulter Modification des fichiers d'unité existants dans le Guide de l'administrateur système Red Hat Enterprise Linux 7.

Utilisation de SSM Agent sur des instances EC2 pour macOS

AWS Systems Manager (SSM Agent) traite les demandes de Systems Manager et configure votre machine comme indiqué dans la demande. Utilisez les procédures suivantes pour installer, configurer ou désinstaller l'SSM Agent pour macOS.

> **Note**
>
> L'SSM Agent est préinstallé par défaut sur Amazon Machine Images (AMIs) pour macOS. Vous n'avez pas besoin d'installer l'SSM Agent sur une instance Amazon Elastic Compute Cloud (Amazon EC2) pour macOS sauf si vous l'avez désinstallé.

Le code source de SSM Agent est disponible sur GitHubafin que vous puissiez adapter l'agent 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 SSM Agent, consultez les notes de mise à jour.

Avant d'installer manuellement SSM Agent sur un système d'exploitation macOS, veuillez consulter les informations suivantes.

- SSM Agent est installé par défaut sur les instances EC2 et les Amazon Machine Images suivantes :
  - macOS 10.14.x (Mojave)
  - macOS 10.15.x (Catalina)
  - macOS11,x () BigSur
  - macOS 12.x (Monterey)
  - macOS13,x (Ventura)
  - macOS14,x (Sonoma)

  L'SSM Agent n'a pas besoin d'être installé manuellement surmacOSEC2, sauf s'il a été désinstallé.
- Les instances EC2 pour ne macOS sont pas toutes Régions AWS prises en charge. Pour obtenir la liste des régions dans lesquelles les instances x86 et M1 EC2 pour macOS sont prises en charge, veuillez consulter la rubrique Charges de travail macOS dans les Questions fréquentes Amazon EC2.
- Une nouvelle version de SSM Agent est lancée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Le fait de ne pas utiliser la dernière version de l'agent peut empêcher votre nœud géré d'utiliser diverses capacités et fonctionnalités de Systems Manager. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir SSM Agent à jour sur vos machines. Pour plus d'informations, consultez Automatisation des mises à jour de l'SSM Agent. Abonnez-vous à la page des notes de SSM Agent publication GitHub pour recevoir des notifications concernant les SSM Agent mises à jour.

Rubriques

- Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour macOS

Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour macOS

Connectez-vous à votre instance macOS et effectuez les opérations suivantes pour installer AWS Systems Manager Agent (SSM Agent). Effectuez ces étapes sur chaque instance qui exécutera les commandes avec Systems Manager. Les commandes fournies dans cette procédure peuvent également être transmises aux instances Amazon EC2 sous forme de scripts via les données utilisateurs.

Pour installer l'SSM Agent sur macOS

1. Téléchargez le fichier d'installation de l'agent pour les instances x86_64 à l'aide de la commande suivante.

   Dans la commande ci-après, remplacez *region* (région) par vos propres informations. Pour obtenir une liste des valeurs *region* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la Référence générale d'Amazon Web Services.

   ```
   sudo wget https://s3.region.amazonaws.com/amazon-ssm-region/latest/darwin_amd64/
   amazon-ssm-agent.pkg
   ```

   Par Apple silicon exemple, utilisez la commande suivante.

   ```
   sudo wget https://s3.region.amazonaws.com/amazon-ssm-region/latest/darwin_arm64/
   amazon-ssm-agent.pkg
   ```

   Voici un exemple.

   ```
   sudo wget https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/latest/
   darwin_amd64/amazon-ssm-agent.pkg
   ```

2. Utilisez la commande suivante pour exécuter le programme d'installation de l'SSM Agent.

   x86_64:

   ```
   sudo installer -pkg amazon-ssm-agent.pkg -target /
   ```

3. Vérifiez le statut de l'agent.

   Pour déterminer si l'SSM Agent est en cours d'exécution, consultez le journal de l'agent sur `/var/log/amazon/ssm/amazon-ssm-agent.log`.

4. Exécutez la commande suivante pour démarrer le service si le journal de l'agent indique que « amazon-ssm-agent est arrêté ».

```
sudo launchctl load -w /Library/LaunchDaemons/com.amazon.aws.ssm.plist && sudo
 launchctl start com.amazon.aws.ssm
```

> **Important**
>
> Une nouvelle version de SSM Agent est lancée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Le fait de ne pas utiliser la dernière version de l'agent peut empêcher votre nœud géré d'utiliser diverses capacités et fonctionnalités de Systems Manager. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir SSM Agent à jour sur vos machines. Pour plus d'informations, consultez Automatisation des mises à jour de l'SSM Agent. Abonnez-vous à la page des notes de SSM Agent publication GitHub pour recevoir des notifications concernant les SSM Agent mises à jour.

Désinstaller SSM Agent sur les instances macOS

macOS ne prend pas en charge la désinstallation de fichiers PKG, en mode natif. Pour désinstaller AWS Systems Manager Agent (SSM Agent) d'une instance Amazon Elastic Compute Cloud (Amazon EC2) macOS pour, vous pouvez utiliser AWS le script géré depuis l'emplacement suivant.

https://github.com/aws/amazon-ssm-agent/blob/mainline/Tools/src/update/darwin/uninstall.sh

Utilisation de SSM Agent sur des instances EC2 pour Windows Server

AWS Systems Manager L'agent (SSM Agent) est préinstallé, par défaut, sur Windows Server les Amazon Machine Images (AMIs) fournis par AWS. La prise en charge est fournie pour les versions suivantes du système d'exploitation (OS).

- Les AMIs Windows Server 2008-2012 R2 publiées en novembre 2016 ou après
- Windows Server 2016, 2019 et 2022

Notes de support pour les versions précédentes

Les AMIs Windows Server publiées avant novembre 2016 utilisent le service EC2Config pour traiter les demandes et configurer les instances.

À moins que vous n'ayez une raison précise d'utiliser le service EC2Config ou une version antérieure de SSM Agent pour traiter les demandes Systems Manager, nous vous conseillons de télécharger et d'installer la dernière version de SSM Agent sur chacune de vos instances Amazon Elastic Compute Cloud (Amazon EC2) ou machines non EC2 configurées pour Systems Manager dans un environnement hybride et multicloud.

Depuis le 14 janvier 2020, Windows Server 2008 n'est plus pris en charge pour les mises à jour de fonctions ou de sécurité de Microsoft. Les Amazon Machine Images (AMIs) héritées, pour Windows Server 2008 et 2008 R2, incluent toujours la version 2 de l'SSM Agent, mais Systems Manager ne prend plus officiellement en charge les versions 2008 et ne met plus à jour l'agent pour ces versions de Windows Server. En outre, SSM Agent version 3 peut ne pas être compatible avec toutes les opérations sur Windows Server 2008 et 2008 R2. La version finale officiellement prise en charge de l'SSM Agent pour les versions Windows Server 2008 est 2.3.1644.0.

Garder l'SSM Agent à jour

Une nouvelle version de SSM Agent est lancée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Le fait de ne pas utiliser la dernière version de l'agent peut empêcher votre nœud géré d'utiliser diverses capacités et fonctionnalités de Systems Manager. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir SSM Agent à jour sur vos machines. Pour plus d'informations, consultez Automatisation des mises à jour de l'SSM Agent. Abonnez-vous à la page des notes de SSM Agent publication GitHub pour recevoir des notifications concernant les SSM Agent mises à jour.

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

Rubriques

- Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Windows Server
- Configurer l'SSM Agent pour utiliser un proxy pour les instances Windows Server

Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Windows Server

AWS Systems Manager L'agent (SSM Agent) est préinstallé, par défaut, sur les appareils suivants Amazon Machine Images (AMIs) Windows Server fournis par Amazon :

- Les AMIs Windows Server 2008-2012 R2 publiées en novembre 2016 ou après
- Windows Server 2016, 2019 et 2022

Installation SSM Agent sur des instances EC2 pour Windows Server

Si nécessaire, vous pouvez aussi télécharger et installer manuellement la version la plus récente de l'SSM Agent sur votre instance Amazon Elastic Compute Cloud (Amazon EC2) pour Windows Server à l'aide de la procédure suivante. Les commandes fournies dans cette procédure peuvent également être transmises aux instances Amazon EC2 sous forme de scripts via les données utilisateurs.

SSM Agentnécessite Windows PowerShell 3.0 ou version ultérieure pour exécuter certains AWS Systems Manager documents (documents SSM) sur des Windows Server instances (par exemple, le `AWS-ApplyPatchBaseline` document existant). Vérifiez que vos instances Windows Server exécutent Windows Management Framework 3.0 ou version ultérieure. Ce framework inclut Windows PowerShell. Pour de plus amples informations, consultez Windows Management Framework 3.0.

> **Note**
>
> Cette procédure s'applique à l'installation ou la réinstallation de l'SSM Agent sur une instance EC2 pour Windows Server. Si vous devez installer l'agent sur un serveur local ou une machine virtuelle (VM) afin qu'il puisse être utilisé avec Systems Manager, reportez-vous à la section Comment installer le SSM Agent sur des nœuds Windows hybrides.

Pour installer manuellement la dernière version de l'SSM Agent sur les instances EC2 pour Windows Server

1. Connectez-vous à votre instance à l'aide de Remote Desktop ou de Windows PowerShell. Pour plus d'informations, consultez Connect to your instance dans le guide de l'utilisateur Amazon EC2.
2. Téléchargez la version la plus récente de l'SSM Agent sur votre instance. Vous pouvez télécharger à l'aide de PowerShell commandes ou d'un lien de téléchargement direct.

> **Note**
>
> Les URL de cette étape vous permettent de télécharger à SSM Agent partir de n'importe quel Région AWS site. 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 Région AWS prise en charge par AWS Systems Manager, par exemple `us-east-2` pour la région USA Est (Ohio). Pour obtenir une liste des valeurs *region* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la Référence générale d'Amazon Web Services.

PowerShell

Exécutez les trois PowerShell commandes suivantes dans l'ordre. Ces commandes vous permettent de télécharger l'SSM Agent 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

```
[System.Net.ServicePointManager]::SecurityProtocol = 'TLS12'
$progressPreference = 'silentlyContinue'
Invoke-WebRequest `
    https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/windows_amd64/AmazonSSMAgentSetup.exe `
    -OutFile $env:USERPROFILE\Desktop\SSMAgent_latest.exe
```

32-bit

```
[System.Net.ServicePointManager]::SecurityProtocol = 'TLS12'
$progressPreference = 'silentlyContinue'
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'SSM Agent sur votre instance en utilisant le lien suivant. Si vous le souhaitez, mettez à jour cette URL avec une URL Région AWS spécifique.

https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/windows_amd64/AmazonSSM AgentSetup .exe

Exécutez le fichier `AmazonSSMAgentSetup.exe` téléchargé pour installer SSM Agent.

3. Démarrez ou redémarrez en SSM Agent envoyant la commande suivante PowerShell :

```
Restart-Service AmazonSSMAgent
```

Désinstallez SSM Agent des instances EC2 pour Windows Server

Pour le désinstaller SSM Agent d'une Windows Server instance, ouvrez le Panneau de configuration, Programmes. Choisissez l'option Uninstall a program (Désinstaller un programme). Ouvrez le menu contextuel (clic droit) pour Amazon SSM Agent et choisissez Uninstall (Désinstaller).

Configurer l'SSM Agent 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 vous avez l'intention de l'utiliserSession Manager, notez que les serveurs proxy HTTPS ne sont pas pris en charge.

> **Note**
>
> Depuis le 14 janvier 2020, Windows Server 2008 n'est plus pris en charge pour les mises à jour de fonctions ou de sécurité de Microsoft. Les Amazon Machine Images (AMIs) héritées, pour Windows Server 2008 et 2008 R2, incluent toujours la version 2 de l'SSM Agent, mais Systems Manager ne prend plus officiellement en charge les versions 2008 et ne met plus à jour l'agent pour ces versions de Windows Server. En outre, SSM Agent version 3 peut ne pas être compatible avec toutes les opérations sur Windows Server 2008 et 2008 R2.

La version finale officiellement prise en charge de l'SSM Agent pour les versions Windows Server 2008 est 2.3.1644.0.

Avant de commencer

Avant de configurer SSM Agent pour utiliser un proxy, prenez note des informations importantes suivantes.

Dans la procédure suivante, vous exécutez une commande pour configurer SSM Agent l'utilisation d'un proxy. La commande inclut un `no_proxy` paramètre avec une adresse IP. L'adresse IP est le point de terminaison des services de métadonnées d'instance (IMDS) pour Systems Manager. Si vous ne le spécifiez pas`no_proxy`, les appels à Systems Manager prennent l'identité du service proxy (si la solution de secours IMDSv1 est activée) ou les appels à Systems Manager échouent (si IMDSv2 est appliqué).

- Pour IPv4, spécifie`no_proxy=169.254.169.254`.
- Pour IPv6, spécifie`no_proxy=[fd00:ec2::254]`. L'adresse IPv6 du service de métadonnées d'instance est compatible avec les commandes IMDSv2. L'adresse IPv6 n'est accessible que sur les instances créées sur le système AWS Nitro. Pour plus d'informations, consultez Comment fonctionne le service de métadonnées d'instance version 2 dans le guide de l'utilisateur Amazon EC2.

Pour configurer l'SSM Agent afin d'utiliser un proxy

1. À l'aide de Remote Desktop ou de Windows PowerShell, 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`* par les informations relatives à votre proxy.

```
$serviceKey = "HKLM:\SYSTEM\CurrentControlSet\Services\AmazonSSMAgent"
$keyInfo = (Get-Item -Path $serviceKey).GetValue("Environment")
$proxyVariables = @("http_proxy=hostname:port", "https_proxy=hostname:port",
 "no_proxy=IP address for instance metadata services (IMDS)")

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 SSM Agent 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 SSM Agent, consultez Affichage des journaux SSM Agent.

```
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: hostname:port
2020-02-24 15:31:54 INFO no_proxy: IP address for instance metadata services (IMDS)
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'SSM Agent

1. À l'aide de Remote Desktop ou de Windows PowerShell, connectez-vous à l'instance à configurer.
2. Si vous vous êtes connecté via Remote Desktop, PowerShell lancez-le 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 SSM Agent

Lors de la configuration des paramètres proxy pour l'SSM Agent 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'SSM Agent. 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.

> **Important**
>
> SSM Agent communique à l'aide du protocole HTTPS. Pour cette raison, vous devez configurer le paramètre `HTTPS proxy` en utilisant l'une des options de paramétrage suivantes.

Les paramètres du proxy SSM Agent sont évalués dans l'ordre suivant.

1. Paramètres du Registre AmazonSSMAgent (`HKLM:\SYSTEM\CurrentControlSet\Services\AmazonSSMAgent`)
2. Variables d'environnement du système (`http_proxy`, `https_proxy`, `no_proxy`)
3. LocalSystem variables d'environnement du compte utilisateur `http_proxyhttps_proxy`,`no_proxy`)
4. Paramètres d'Internet Explorer (`HTTP`, `secure`, `exceptions`)
5. Paramètres de proxy WinHTTP (`http=`, `https=`, `bypass-list=`)

Paramètres de proxy de l'SSM Agent et services Systems Manager

Si vous avez configuré le SSM Agent pour utiliser un proxy et que vous utilisez AWS Systems Manager des fonctionnalités telles que Run Command etPatch Manager, qui utilisent PowerShell le client Windows Update lors de leur exécution sur des Windows Server instances, configurez des paramètres de proxy supplémentaires. Sinon, l'opération risque d'échouer car les paramètres de proxy utilisés par le client Windows Update PowerShell et le client Windows Update ne sont pas hérités de la configuration du SSM Agent proxy.

Pour Run Command, configurez les paramètres de proxy de `WinINet` sur vos instances Windows Server. Les commandes `[System.Net.WebRequest]` fournies sont spécifiques à une session. Pour appliquer ces configurations aux commandes réseau suivantes exécutées dansRun Command, ces commandes doivent précéder les autres PowerShell commandes dans la même entrée du `aws:runPowershellScript` plugin.

Les PowerShell commandes suivantes renvoient les paramètres de `WinINet` proxy actuels et appliquent vos paramètres de proxy à`WinINet`.

```
[System.Net.WebRequest]::DefaultWebProxy

$proxyServer = "http://hostname:port"
$proxyBypass = "169.254.169.254"
$WebProxy = New-Object System.Net.WebProxy($proxyServer,$true,$proxyBypass)

[System.Net.WebRequest]::DefaultWebProxy = $WebProxy
```

Pour Patch Manager, vous devez configurer les paramètres de proxy à l'échelle du système, afin que le client Windows Update puisse rechercher et télécharger des mises à jour. Nous vous recommandons d'utiliser Run Command 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 proxy-server="hostname:port" bypass-list="169.254.169.254"
```

Pour plus d'informations sur l'utilisation de Run Command, consultez AWS Systems Manager Run Command.

Vérification du statut de l'SSM Agent et démarrage de l'agent

Cette rubrique répertorie les commandes permettant de vérifier si AWS Systems Manager Agent (SSM Agent) est exécuté sur chaque système d'exploitation pris en charge. Elle fournit aussi les commandes nécessaires pour démarrer l'agent s'il n'est pas en cours d'exécution.

Système d'exploitation	Commande pour vérifier le statut de l'SSM Agent	Commande pour démarrer l'SSM Agent
Amazon Linux 1	`sudo status amazon-ssm-agent`	`sudo start amazon-ssm-agent`
Amazon Linux 2 et Amazon Linux 2023	`sudo systemctl status amazon-ssm-agent`	`sudo systemctl enable amazon-ssm-agent` `sudo systemctl start amazon-ssm-agent`

Système d'exploitation	Commande pour vérifier le statut de l'SSM Agent	Commande pour démarrer l'SSM Agent
CentOS 6.x	`sudo status amazon-ssm-agent`	`sudo start amazon-ssm-agent`
CentOS 7.x et CentOS 8.x	`sudo systemctl status amazon-ssm-agent`	`sudo systemctl enable amazon-ssm-agent` `sudo systemctl start amazon-ssm-agent`
Debian Server 8, 9 et 10	`sudo systemctl status amazon-ssm-agent`	`sudo systemctl enable amazon-ssm-agent` `sudo systemctl start amazon-ssm-agent`
macOS	Vérification du fichier journal de l'agent à `/var/log/amazon/ssm/amazon-ssm-agent.log`	`sudo launchctl load -w /Library/LaunchDaemons/com.amazon.aws.ssm.plist` `sudo launchctl start com.amazon.aws.ssm`
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`

Système d'exploitation	Commande pour vérifier le statut de l'SSM Agent	Commande pour démarrer l'SSM Agent
Red Hat Enterprise Linux(RHEL) 7.x, 8.x et 9.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 Server 14.04 (tous) et 16.04 (32 bits)	`sudo status amazon-ssm-agent`	`sudo start amazon-ssm-agent`
Instances Ubuntu Server 16.04 64 bits (installation du package deb)	`sudo systemctl status amazon-ssm-agent`	`sudo systemctl enable amazon-ssm-agent` `sudo systemctl start amazon-ssm-agent`
Ubuntu Server 16.04, 18.04 et 20.04 LTS, 20.10 STR 64 bits et 22.04 LTS (installation du package Snap)	`sudo systemctl status snap.amazon-ssm-agent.amazon-ssm-agent.service`	`sudo snap start amazon-ssm-agent`
Windows Server	Exécuter dans PowerShell : `Get-Service AmazonSSMAgent`	Exécuter en mode PowerShell administrateur : `Start-Service AmazonSSMAgent`

Plus d'informations

- Utilisation de SSM Agent sur des instances EC2 pour Linux
- Utilisation de SSM Agent sur des instances EC2 pour Windows Server

Vérification du numéro de version de l'SSM Agent

Certaines AWS Systems Manager fonctionnalités sont soumises à des prérequis, notamment l'installation d'une version minimale de Systems Manager Agent (SSM Agent) sur vos nœuds gérés. Vous pouvez connaître la version de l'SSM Agent actuellement installée sur vos nœuds gérés à l'aide de la console Systems Manager ou en vous connectant à vos nœuds gérés.

Les procédures suivantes décrivent comment connaître la version de l'SSM Agent actuellement installée sur vos nœuds gérés.

Pour vérifier le numéro de version de l'SSM Agent installé sur un nœud géré

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Dans la colonne Version de l'SSM Agent, notez le numéro de la Version de l'agent.

Pour obtenir la version de l'SSM Agent actuellement installée à partir du système d'exploitation

Sélectionnez parmi les onglets suivants pour obtenir la version SSM Agent actuellement installée à partir d'un système d'exploitation.

Amazon Linux 1, Amazon Linux 2, and Amazon Linux 2023

> **Note**
>
> Cette commande varie en fonction du gestionnaire de package de votre système d'exploitation.

1. Connectez-vous à votre nœud géré.
2. Exécutez la commande suivante.

   ```
   yum info amazon-ssm-agent
   ```

   La sortie générée lors de l'exécution de cette commande est semblable à ce qui suit.

```
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Installed Packages
Name        : amazon-ssm-agent
Arch        : x86_64
Version     : 3.0.655.0
```

CentOS

1. Connectez-vous à votre nœud géré.
2. Exécutez la commande suivante pour CentOS 6 et 7.

   ```
   yum info amazon-ssm-agent
   ```

   La sortie générée lors de l'exécution de cette commande est semblable à ce qui suit.

   ```
   Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
   Installed Packages
   Name        : amazon-ssm-agent
   Arch        : x86_64
   Version     : 3.0.655.0
   ```

Debian Server

1. Connectez-vous à votre nœud géré.
2. Exécutez la commande suivante.

   ```
   apt list amazon-ssm-agent
   ```

   La sortie générée lors de l'exécution de cette commande est semblable à ce qui suit.

   ```
   apt list amazon-ssm-agent
   Listing... Done
   amazon-ssm-agent/now 3.0.655.0-1 amd64 [installed,local]

   3.0.655.0 is the version of SSM agent
   ```

macOS

1. Connectez-vous à votre nœud géré.
2. Exécutez la commande suivante.

```
pkgutil --pkg-info com.amazon.aws.ssm
```

RHEL

1. Connectez-vous à votre nœud géré.
2. Exécutez la commande suivante pour RHEL 6, 7, 8 et 9.

```
yum info amazon-ssm-agent
```

La sortie générée lors de l'exécution de cette commande est semblable à ce qui suit.

```
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Installed Packages
Name        : amazon-ssm-agent
Arch        : x86_64
Version     : 3.0.655.0
```

Exécutez la commande suivante pour l'utilitaire de package DNF.

```
dnf info amazon-ssm-agent
```

SLES

1. Connectez-vous à votre nœud géré.
2. Exécutez la commande suivante pour SLES 12 et 15.

```
zypper info amazon-ssm-agent
```

La sortie générée lors de l'exécution de cette commande est semblable à ce qui suit.

```
Loading repository data...
Reading installed packages...
```

```
Information for package amazon-ssm-agent:
----------------------------------------
Repository : @System
Name : amazon-ssm-agent
Version : 3.0.655.0-1
```

Ubuntu Server

> **Note**
>
> Pour vérifier si votre instance Ubuntu Server 16.04 utilise des packages deb ou Snap, consultez Installation manuelle de SSM Agent sur les instances Ubuntu Server.

1. Connectez-vous à votre nœud géré.
2. Exécutez la commande suivante pour Ubuntu Server 16.04 et 14.04 64 bits (avec un package d'installation deb).

   ```
   apt list amazon-ssm-agent
   ```

   La sortie générée lors de l'exécution de cette commande est semblable à ce qui suit.

   ```
   apt list amazon-ssm-agent
   Listing... Done
   amazon-ssm-agent/now 3.0.655.0-1 amd64 [installed,local]

   3.0.655.0 is the version of SSM agent
   ```

   Exécutez la commande suivante pour les instances Ubuntu Server 22.04 LTS, 20.10 STR et 20.04, 18.04 et 16.04 LTS 64 bits (avec un package Snap).

   ```
   sudo snap list amazon-ssm-agent
   ```

   La sortie générée lors de l'exécution de cette commande est semblable à ce qui suit.

   ```
   snap list amazon-ssm-agent
   Name Version Rev Tracking Publisher Notes
   amazon-ssm-agent 3.0.529.0 3552 latest/stable/… aws# classic-
   ```

```
3.0.529.0 is the version of SSM agent
```

Windows

1. Connectez-vous à votre nœud géré.
2. Exécutez la PowerShell commande suivante.

   ```
   & "C:\Program Files\Amazon\SSM\amazon-ssm-agent.exe" -version
   ```

   La sortie générée lors de l'exécution de cette commande est semblable à ce qui suit.

   ```
   SSM Agent version: 3.1.804.0
   ```

Nous vous recommandons d'utiliser la dernière version de l'SSM Agent afin de bénéficier des fonctionnalités nouvelles ou mises à jour. Pour vous assurer que vos instances gérées exécutent toujours la up-to-date version la plus complète duSSM Agent, vous pouvez automatiser le processus de mise à jour duSSM Agent. Pour plus d'informations, voir Automatisation des mises à jour de l'SSM Agent.

Affichage des journaux SSM Agent

AWS Systems Manager L'agent (SSM Agent) écrit des informations sur les exécutions, les commandes, les actions planifiées, les erreurs et l'état de santé dans les fichiers journaux de chaque nœud géré. Vous pouvez consulter les fichiers journaux en vous connectant manuellement à un nœud géré, ou vous pouvez envoyer automatiquement les journaux à Amazon CloudWatch Logs. Pour plus d'informations sur l'envoi de CloudWatch journaux à Logs, consultezSurveillance AWS Systems Manager.

Vous pouvez afficher les journaux SSM Agent sur les nœuds gérés aux emplacements ci-dessous.

Linux and macOS

```
/var/log/amazon/ssm/
```

Windows

```
%PROGRAMDATA%\Amazon\SSM\Logs\
```

Pour les nœuds gérés Linux, les fichiers SSM Agent `stderr` et `stdout` sont écrits dans le répertoire suivant : `/var/lib/amazon/ssm/`.

Pour les nœuds gérés par Windows, les fichiers SSM Agent `stderr` et `stdout` sont écrits dans le répertoire suivant : `%PROGRAMDATA%\Amazon\SSM\InstanceData\`.

Pour de plus amples informations sur l'activation de la journalisation de débogage de SSM Agent, veuillez consulter Autorisation de la journalisation de débogage de l'SSM Agent.

Pour plus d'informations sur `cihub/seelog` la configuration, consultez le wiki Seelog sur. GitHub Pour des exemples de `cihub/seelog` configurations, consultez le référentiel d'exemples cihub/seelog sur. GitHub

Autorisation de la journalisation de débogage de l'SSM Agent

Utilisez la procédure suivante pour activer la journalisation de débogage SSM Agent sur vos nœuds gérés.

Linux and macOS

Pour autoriser la journalisation de débogage de l'SSM Agent sur les nœuds gérés Linux et macOS

1. Utilisez une fonctionnalité de Session Manager AWS Systems Manager, pour vous connecter au nœud géré sur lequel vous souhaitez autoriser la journalisation du débogage, ou connectez-vous au nœud géré. Pour plus d'informations, consultez Utilisation des Session Manager.
2. Localisez le fichier seelog.xml.template

   Linux :

   Sur la plupart des types de nœud géré Linux, le fichier se trouve 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 doit être créé dans le répertoire `/snap/amazon-ssm-agent/current/seelog.xml.template`. Copiez ce fichier du répertoire `/snap/amazon-ssm-agent/current/` vers le répertoire `/etc/amazon/ssm/` avant d'apporter des modifications.

   macOS:

Sur les types d'instance macOS, le fichier se trouve dans le répertoire `/opt/aws/ssm/seelog.xml.template`.

3. Remplacez le nom du fichier `seelog.xml.template` par `seelog.xml`.

> **Note**
>
> Sur Ubuntu Server 20.10 STR & 20.04, 18.04 et 16.04 LTS, le fichier `seelog.xml` doit être créé dans le répertoire `/etc/amazon/ssm/`. Vous pouvez créer ce répertoire et ce fichier en exécutant les commandes suivantes.
>
> ```
> sudo mkdir -p /etc/amazon/ssm
> ```
>
> ```
> sudo cp -p /snap/amazon-ssm-agent/current/seelog.xml.template /etc/amazon/ssm/seelog.xml
> ```

4. 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">
   ```

5. (Facultatif) Redémarrez SSM Agent à l'aide de la commande suivante.

   Linux :

   ```
   sudo service amazon-ssm-agent restart
   ```

   macOS:

   ```
   sudo /opt/aws/ssm/bin/amazon-ssm-agent restart
   ```

Windows

Pour autoriser la journalisation de débogage de l'SSM Agent sur les nœuds gérés Windows Server

1. Utilisez Session Manager pour vous connecter au nœud géré sur lequel vous souhaitez activer la journalisation de débogage, ou connectez-vous aux nœuds gérés. Pour plus d'informations, consultez Utilisation des Session Manager.
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\{{EXECUTABLENAME}}.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 SSM Agent à l'aide de PowerShell la commande suivante en mode administrateur.

   ```
   Restart-Service AmazonSSMAgent
   ```

Limitation de l'accès aux commandes de niveau racine via l'SSM Agent

AWS Systems Manager Agent (SSM Agent) s'exécute sur les instances Amazon Elastic Compute Cloud (Amazon EC2) et sur d'autres types de machines dans des environnements hybrides et multicloud à l'aide des autorisations root (Linux) ou des 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'SSM Agent possède des autorisations racine ou SYSTEM. (Dans AWS, une entité de confiance qui peut effectuer des actions et accéder à des ressources AWS est appelée principale. Un principal peut être un Utilisateur racine d'un compte AWS, un 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'SSM Agent, mais il permet également à un principal d'exécuter des codes malveillants en exploitant d'éventuelles vulnérabilités dans l'SSM Agent.

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 principaux dans votre organisation. Toutefois, nous vous recommandons de renforcer votre posture de sécurité en limitant les nœuds gérés sur lesquels un principal peut exécuter ses commandes. Cela est possible dans la politique IAM affectée au principal. Dans la politique IAM, vous pouvez inclure une condition qui limite l'utilisateur à l'exécution de commandes uniquement sur les nœuds gérés balisés à l'aide de balises spécifiques ou de combinaisons de balises.

Par exemple, supposons que vous avez deux parcs de serveurs, l'un à des fins de test, l'autre à des fins de production. Dans la politique IAM 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` ou `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 politique 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, consultez les rubriques suivantes :

- Restriction de l'accès Run Command en fonction des balises
- Limiter l'accès à la session en fonction des balises d'instance

Automatisation des mises à jour de l'SSM Agent

AWS publie une nouvelle version d' AWS Systems Manager Agent (SSM Agent) lorsque nous ajoutons ou mettons à jour les fonctionnalités de Systems Manager. Si vos nœuds gérés 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'SSM Agent sur vos nœuds gérés à l'aide de l'une des méthodes suivantes.

Mises à jour des agents sur le système d'exploitation Bottlerocket

SSM Agent sur le système d'exploitation Bottlerocket ne peut pas être mis à jour à l'aide du document de commande Systems Manager `AWS-UpdateSSMAgent`. Les mises à jour sont gérées dans le conteneur de contrôle Bottlerocket. Pour plus d'informations, consultez Bottlerocket Control Container et Bottlerocket Update infrastructure on. GitHub

Exigence de version macOS

Si une instance exécute macOS version 11.0 (Big Sur) ou ultérieure, elle doit disposer de SSM Agent version 3.1.941.0 ou supérieure pour exécuter le document AWS-UpdateSSMAgent. Si l'instance exécute une version de l'SSM Agent publiée avant la version 3.1.941.0, mettez à jour votre SSM Agent pour exécuter l'AWS-UpdateSSMAgent en exécutant les commandes `brew update` et `brew upgrade amazon-ssm-agent`.

Méthode	Détails
Mise à jour automatisée en un clic sur tous les nœuds gérés (recommandé)	Vous pouvez configurer tous les nœuds gérés de votre Compte AWS ordinateur pour vérifier et télécharger automatiquement les nouvelles versions deSSM Agent. Pour ce faire, choisisse z Mise à jour automatique de l'SSM Agent dans l'onglet Paramètres dans Fleet Manager, comme décrit plus loin dans cette rubrique.
Mise à jour globale ou sélective	Vous pouvez utiliser State Manager une fonctionnalité de AWS Systems Manager pour créer une association qui se télécharge et s'installe automatiquement SSM Agent sur vos nœuds gérés. Si vous souhaitez limiter

Méthode	Détails
	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 nœuds gérés qui seront mis à jour. Pour de plus amples informations sur la création d'une association State Manager, veuillez consulter Démonstration : Mise à jour automatique de l'SSM Agent (CLI). Pour plus d'informations sur l'utilisation d'une fenêtre de maintenance, consultez Procédure : Créer une fenêtre de maintenance pour mettre à jour l'SSM Agent (AWS CLI) et Démonstration : Créer une fenêtre de maintenance pour mettre automatiquement à jour l'SSM Agent (console).
Mise à jour globale ou sélective pour les nouveaux environnements	Si vous débutez avec Systems Manager, nous vous recommandons d'utiliser l'option Update Systems Manager (SSM) Agent toutes les deux semaines dans Quick Setup une fonctionnalité de AWS Systems Manager. Quick Setupvous permet de créer une configuration de mise à jour globale pour tous vos nœuds gérés ou de choisir de manière sélective les nœuds gérés à mettre à jour. Pour plus d'informations, consultez Gestion des hôtes Amazon EC2.

Si vous préférez mettre à SSM Agent jour vos nœuds gérés manuellement, vous pouvez vous abonner aux notifications AWS publiées lorsqu'une nouvelle version de l'agent est publiée. Pour plus d'informations, veuillez consulter Abonnement aux notifications SSM Agent. Après vous être abonné aux notifications, vous pouvez utiliser l'option Run Command pour mettre à jour manuellement une ou

plusieurs nœuds gérés à la dernière version. Pour plus d'informations, consultez Mise à jour de SSM Agent à l'aide de Run Command.

Mise à jour automatique de l'SSM Agent

Vous pouvez configurer Systems Manager de sorte à mettre à jour l'SSM Agent automatiquement sur tous les nœuds gérés Linux et Windows dans votre 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, Systems Manager met 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 nœuds gérés exécutent toujours la up-to-date version la plus complète deSSM Agent.

> **Note**
>
> Si vous utilisez une commande `yum` pour mettre à jour l'SSM Agent sur un nœud géré après que l'agent a été installé ou mis à jour en utilisant le document SSM `AWS-UpdateSSMAgent`, le message suivant peut s'afficher : « Warning: RPMDB altered outside of yum (Avertissement : RPMDB modifié en dehors de yum) ». Ce message est prévu pour s'afficher. Il peut être ignoré sans risque.

Pour mettre à jour automatiquement l'SSM Agent

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Sélectionnez l'onglet Settings.
4. Dans la zone de Mise à jour automatique de l'agent, choisissez Mise à jour automatique de l'SSM Agent.

Pour modifier la version de SSM Agent vers laquelle votre flotte se met à jour, choisissez Edit (Modifier) dans Agent auto update (Mise à jour automatique de l'agent) dans l'onglet Settings (Paramètres). Ensuite, saisissez le numéro de version de l'SSM Agent à laquelle vous voulez faire la mise à jour dans Version sous Settings (Paramètres). Si vous ne spécifiez pas de version, l'agent est mis à jour avec la dernière version.

Pour arrêter automatiquement le déploiement des versions mises à jour de SSM Agent sur tous les nœuds gérés de votre compte, choisissez Delete (Supprimer) dans Agent auto update (Mise à jour automatique de l'agent) dans l'onglet Settings (Paramètres). Cette action supprime l'association State Manager qui met automatiquement à jour l'SSM Agent sur vos nœuds gérés.

Abonnement aux notifications SSM Agent

Amazon Simple Notification Service (Amazon SNS) peut vous avertir lorsque de nouvelles versions AWS Systems Manager d'Agent SSM Agent () sont publiées. Pour vous abonner à ces notifications, utilisez la procédure suivante.

> **Tip**
>
> Vous pouvez également vous abonner aux notifications en consultant la page des notes de SSM Agent publication surGitHub.

Pour s'abonner aux notifications SSM Agent

1. Ouvrez la console Amazon SNS à partir de l'adresse https://console.aws.amazon.com/sns/v3/home.
2. Dans le sélecteur de région dans la barre de navigation, sélectionnez US East (N. Virginia) (USA Est [Virginie du Nord]), si elle n'est pas déjà sélectionnée. Vous devez sélectionner cette option Région AWS car les notifications Amazon SNS pour SSM Agent lesquelles vous vous abonnez sont générées uniquement à partir de cette région.
3. Dans le panneau de navigation, sélectionnez Abonnements.
4. Sélectionnez Créer un abonnement.
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, sélectionnez `Email` ou `SMS`.

    c. Pour Enpoint (Point de terminaison), selon que vous avez choisi `Email` ou `SMS` à l'étape précédente, entrez une adresse e-mail ou un indicatif régional et un numéro pour recevoir des notifications.

    d. Sélectionnez Créer un abonnement.

6. Si vous sélectionnez `Email`, vous recevrez un e-mail vous demandant de confirmer votre abonnement. Ouvrez le message et suivez les instructions pour terminer votre abonnement.

Chaque fois qu'une nouvelle version de SSM Agent 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 SSM Agent

1. Ouvrez la console Amazon SNS.
2. Dans le panneau de navigation, sélectionnez Abonnements.
3. Sélectionnez les abonnements, puis choisissez Delete (Supprimer). Lorsque vous êtes invité à confirmer l'opération, choisissez Supprimer.

Résolution des problèmes de SSM Agent

Si vous rencontrez des problèmes lors de l'exécution des opérations sur vos nœuds gérés, il se peut qu'il y ait un problème avec AWS Systems Manager Agent (SSM Agent). Utilisez les informations suivantes pour vous permettre de visualiser les fichiers journaux de l'SSM Agent et dépanner l'agent.

Rubriques

L'SSM Agent est obsolète

Une nouvelle version de SSM Agent est lancée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Le fait de ne pas utiliser la dernière version de l'agent peut empêcher votre nœud géré d'utiliser diverses capacités et fonctionnalités de Systems Manager. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir SSM Agent à jour sur vos machines. Pour plus d'informations, consultez Automatisation des mises à jour de l'SSM Agent. Abonnez-vous à la page

des notes de SSM Agent publication GitHub pour recevoir des notifications concernant les SSM Agent mises à jour.

Résolution des problèmes à l'aide des fichiers journaux SSM Agent

SSM Agent consigne des informations dans les fichiers suivants. Les informations de ces fichiers peuvent également vous aider à résoudre les problèmes. Pour de plus amples informations sur les fichiers journaux de l'SSM Agent, notamment l'activation de la journalisation du débogage, veuillez consulter Affichage des journaux SSM Agent.

> **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 et macOS

- `/var/log/amazon/ssm/amazon-ssm-agent.log`
- `/var/log/amazon/ssm/errors.log`

Pour les nœuds gérés Linux, vous pouvez trouver de plus amples informations dans le fichier `messages` rédigé dans le répertoire suivant : `/var/log`.

Pour en savoir plus sur le dépannage à l'aide des journaux d'agents, consultez la rubrique Comment puis-je utiliser les journaux SSM Agent pour résoudre des problèmes liés à SSM Agent dans mon instance gérée ? dans le Centre de connaissances AXS re:Post AWS .

Les fichiers journaux de l'agent ne tournent pas (Windows)

Si vous spécifiez une rotation des fichiers journaux basée sur la date dans le fichier seelog.xml (sur les nœuds gérés Windows Server) et que les journaux ne tournent pas, spécifiez le paramètre

`fullname=true`. Voici un exemple de fichier de configuration seelog.xml pour lequel le paramètre `fullname=true` est spécifié.

```
<seelog type="adaptive" mininterval="2000000" maxinterval="100000000"
 critmsgcount="500" minlevel="debug">
   <exceptions>
      <exception filepattern="test*" minlevel="error" />
   </exceptions>
   <outputs formatid="fmtinfo">
      <console formatid="fmtinfo" />
      <rollingfile type="date" datepattern="200601021504" maxrolls="4" filename="C:
\ProgramData\Amazon\SSM\Logs\amazon-ssm-agent.log" fullname=true />
      <filter levels="error,critical" formatid="fmterror">
         <rollingfile type="date" datepattern="200601021504" maxrolls="4" filename="C:
\ProgramData\Amazon\SSM\Logs\errors.log" fullname=true />
      </filter>
   </outputs>
   <formats>
      <format id="fmterror" format="%Date %Time %LEVEL [%FuncShort @ %File.%Line] %Msg
%n" />
      <format id="fmtdebug" format="%Date %Time %LEVEL [%FuncShort @ %File.%Line] %Msg
%n" />
      <format id="fmtinfo" format="%Date %Time %LEVEL %Msg%n" />
   </formats>
</seelog>
```

Impossible de se connecter aux points de terminaison SSM

SSM Agent doit autoriser le trafic sortant HTTPS (port 443) vers les points de terminaison suivants :

- `ssm.`*`region`*`.amazonaws.com`
- `ssmmessages.`*`region`*`.amazonaws.com`

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

> **Note**
>
> Avant 2024, `ec2messages.region.amazonaws.com` elle était également requise. Pour les Régions AWS lancements effectués avant 2024, l'autorisation du trafic `ssmmessages.region.amazonaws.com` est toujours obligatoire mais facultativeec2messages.region.amazonaws.com.
> Pour les régions lancées à partir de 2024, il `ssmmessages.region.amazonaws.com` est nécessaire d'autoriser le trafic à destination, mais les `ec2messages.region.amazonaws.com` terminaux ne sont pas pris en charge pour ces régions.

SSM Agentne fonctionnera pas s'il ne peut pas communiquer avec les points de terminaison précédents, comme décrit, même si vous utilisez AWS provided Amazon Machine Images (AMIs) tel qu'Amazon Linux 2 ou Amazon Linux 2023. Votre configuration réseau doit avoir un accès Internet ouvert, ou bien des points de terminaison de cloud privé virtuel (VPC) personnalisés doivent être configurés. Si vous ne prévoyez pas de créer un point de terminaison de VPC personnalisé, vérifiez vos passerelles Internet ou NAT. Pour plus d'informations sur la gestion des points de terminaison de VPC, consultez Améliorez la sécurité des instances EC2 en utilisant des points de terminaison VPC pour Systems Manager.

Utilisation de `ssm-cli` pour résoudre des problèmes de disponibilité des nœuds gérés

À partir de la version 3.1.501.0 de l'SSM Agent, vous pouvez l'utiliser `ssm-cli` pour déterminer si un nœud géré répond aux exigences principales pour être géré par Systems Manager et pour apparaître dans les listes de nœuds gérés dans Fleet Manager. `ssm-cli` est un outil de ligne de commande autonome inclus dans l'installation SSM Agent. Il contient des commandes préconfigurées qui rassemblent les informations requises afin de déterminer pourquoi une instance Amazon EC2 ou une machine non EC2, confirmée comme en cours d'exécution, ne figure pas dans vos listes de nœuds gérées dans Systems Manager. Ces commandes sont exécutées lorsque vous spécifiez l'option `get-diagnostics`.

Pour plus d'informations, voir Résolution des problèmes de disponibilité des nœuds gérés en utilisant `ssm-cli`.

AWS Systems Manager Quick Setup

UtilisezQuick Setup, une fonctionnalité de AWS Systems Manager, pour configurer rapidement les services et fonctionnalités Amazon Web Services fréquemment utilisés conformément aux meilleures pratiques recommandées. Quick Setupsimplifie la configuration des services, y compris Systems Manager, en automatisant les tâches courantes ou recommandées. Ces tâches incluent, par exemple, la création de rôles de profil d'instance AWS Identity and Access Management (IAM) requis et la mise en place de meilleures pratiques opérationnelles, telles que des analyses de correctifs périodiques et la collecte d'inventaire. Aucun frais d'utilisation Quick Setup. Toutefois, des coûts peuvent être encourus en fonction du type de services configuré et des limites d'utilisation, sans frais pour les services utilisés pour la configuration de votre service. Pour vos premiers pas dans Quick Setup, ouvrez Systems Manager console. Dans le panneau de navigation, sélectionnez Quick Setup.

> **Note**
>
> Si vous avez été dirigé vers Quick Setup pour vous aider dans la configuration de vos instances de sorte qu'elles soient gérées par Systems Manager, suivez la procédure dans Gestion des hôtes Amazon EC2.

Quels sont les avantages d'Quick Setup ?

Les avantages d'Quick Setup sont les suivants :

- Simplification de la configuration des services et des fonctions

  Quick Setup vous guide tout au long de la configuration des bonnes pratiques opérationnelles et déploie ces configurations automatiquement. Le tableau de bord Quick Setup affiche une vue en temps réel du statut de déploiement de votre configuration.

- Déploiement des configurations automatiquement sur plusieurs comptes

  Vous pouvez l'utiliser de Quick Setup manière individuelle Compte AWS ou multiple Comptes AWS et Régions AWS en intégrant à AWS Organizations. L'utilisation de Quick Setup sur plusieurs comptes garantit le maintien de configurations cohérentes dans votre organisation.

- Élimination de la dérive de configuration

Une dérive de configuration se produit chaque fois qu'une modification apportée par un utilisateur à un service ou une fonction entre en conflit avec les sélections effectuées via Quick Setup. Quick Setup vérifie périodiquement la dérive de configuration et tente de la corriger.

À qui est destiné Quick Setup ?

Quick Setup convient particulièrement aux clients déjà familiarisés avec les services et fonctions qu'ils configurent et désireux de simplifier leur processus d'installation. Si vous ne connaissez pas le service avec lequel Service AWS vous effectuez la configurationQuick Setup, nous vous recommandons d'en savoir plus sur le service. Étudiez le contenu du Guide de l'utilisateur correspondant avant de créer une configuration avec Quick Setup.

Disponibilité de Quick Setup dans les Régions AWS

Dans ce qui suit Régions AWS, vous pouvez utiliser tous les types de Quick Setup configuration pour l'ensemble d'une organisation, tels que configurés dans AWS Organizations, ou uniquement pour les comptes d'organisation et les régions que vous choisissez. Vous pouvez également l'utiliser Quick Setup avec un seul compte dans ces régions.

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

- Europe (Londres)
- Europe (Paris)
- Amérique du Sud (São Paulo)

Dans les régions suivantes, seul le type de configuration Gestion des hôtes est disponible pour les comptes individuels :

- Europe (Milan)
- Asie-Pacifique (Hong Kong)
- Moyen-Orient (Bahreïn)
- Chine (Beijing)
- Chine (Ningxia)
- AWS GovCloud (USA Est)
- AWS GovCloud (US-Ouest)

Pour obtenir une liste de toutes les régions prises en charge pour Systems Manager, consultez la colonne Région dans Points de terminaison de service Systems Manager dans le Référence générale d'Amazon Web Services.

Démarrer avec Quick Setup

Utilisez les informations de cette rubrique pour vous aider à vous préparer à utiliser Quick Setup.

Rubriques

- Configuration de la Région AWS d'accueil
- Rôles et autorisations IAM pour l'intégration de Quick Setup

Configuration de la Région AWS d'accueil

Pour commencerQuick Setup, une fonctionnalité de AWS Systems Manager, vous devez choisir une maison, Région AWS puis embarquer avecQuick Setup. C'est dans la région d'origine que sont créées les AWS ressources utilisées pour déployer vos configurations. Quick Setup Une fois sélectionnée, la région d'accueil ne peut pas être modifiée.

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Quick Setup.
3. Pour Choisir une région d'origine, choisissez l' Région AWS endroit où vous Quick Setupsouhaitez créer les AWS ressources utilisées pour déployer vos configurations.
4. Sélectionnez Get started (Démarrer).

Pour commencer à utiliser Quick Setup, sélectionnez un service ou une fonction dans la liste des types de configuration disponibles. Un type de configuration Quick Setup est spécifique à une fonctionnalité Service AWS ou. Lorsque vous sélectionnez un type de configuration, vous sélectionnez les options à configurer pour ce service ou cette fonction. Par défaut, les types de configuration vous aident à configurer le service ou la fonction de sorte à utiliser les bonnes pratiques recommandées.

Après avoir configuré une configuration, vous pouvez en afficher les détails et le statut de déploiement dans les unités organisationnelles (UO) et les régions. Vous pouvez également consulter le statut de State Manager l'association pour la configuration. State Managerest une capacité de AWS Systems Manager. Dans le panneau Détails de configuration, vous pouvez consulter un résumé de la configuration Quick Setup. Ce résumé inclut des détails sur tous les comptes et toute dérive de configuration détectée.

Rôles et autorisations IAM pour l'intégration de Quick Setup

Lors de l'intégration, Quick Setup crée les rôles AWS Identity and Access Management (IAM) suivants en votre nom :

- `AWS-QuickSetup-StackSet-Local-ExecutionRole` : octroie des autorisations AWS CloudFormation pour utiliser n'importe quel modèle.
- `AWS-QuickSetup-StackSet-Local-AdministrationRole`— Accorde l'autorisation AWS CloudFormation d'assumer`AWS-QuickSetup-StackSet-Local-ExecutionRole`.

Si vous créez un compte de gestion, le compte dans lequel vous créez une organisation, crée Quick Setup également les rôles suivants en AWS Organizations votre nom :

- `AWS-QuickSetup-SSM-RoleForEnablingExplorer` : octroie des autorisations au runbook Automation `AWS-EnableExplorer`. Le `AWS-EnableExplorer` runbook configureExplorer, une

fonctionnalité de Systems Manager, pour afficher des informations pour plusieurs Comptes AWS et. Régions AWS

- `AWSServiceRoleForAmazonSSM`— Un rôle lié à un service qui donne accès aux AWS ressources gérées et utilisées par Systems Manager.
- `AWSServiceRoleForAmazonSSM_AccountDiscovery`— Rôle lié à un service qui autorise Systems Manager à appeler pour découvrir des Compte AWS informations lors Services AWS de la synchronisation des données. Pour plus d'informations, consultez À propos du rôle `AWSServiceRoleForAmazonSSM_AccountDiscovery`.

Lors de l'intégration d'un compte de gestion, Quick Setup permet un accès fiable entre AWS Organizations et CloudFormation pour déployer des Quick Setup configurations au sein de votre organisation. Pour activer l'accès approuvé, votre compte de gestion doit disposer d'autorisations d'administrateur. Après l'intégration, vous n'avez plus besoin d'autorisations d'administrateur. Pour de plus amples informations, consultez Activer l'accès approuvé avec Organizations.

Pour plus d'informations sur les types de AWS Organizations comptes, consultez AWS Organizations la terminologie et les concepts du Guide de AWS Organizations l'utilisateur.

> **Note**
>
> Quick Setuputilise AWS CloudFormation StackSets pour déployer vos configurations dans toutes Comptes AWS les régions. Si le nombre de comptes cibles multiplié par le nombre de régions dépasse 10 000, le déploiement de la configuration échoue. Nous vous recommandons de revoir votre cas d'utilisation et de créer des configurations qui utilisent moins de cibles pour s'adapter à la croissance de votre organisation. Les instances de piles ne sont pas déployées sur le compte de gestion de votre organisation. Pour plus d'informations, consultez Considérations lors de la création d'un ensemble de piles avec des autorisations gérées par le service.

Si votre utilisateur, groupe ou rôle IAM a accès aux opérations API répertoriées dans le tableau suivant, vous pouvez utiliser l'intégralité des fonctions de Quick Setup Il existe deux onglets d'opérations d'API, le premier onglet correspond aux autorisations requises par tous les comptes et le second contient les autorisations supplémentaires dont vous avez besoin pour le compte de gestion de votre organisation.

Non-management account

```
"iam:CreateRole",
"iam:AttachRolePolicy",
"iam:PutRolePolicy",
"iam:GetRole",
"iam:ListRoles",
"iam:PassRole"
"ssm:ListAssociations",
"ssm:ListDocuments",
"ssm:GetDocument",
"ssm:DescribeAssociation",
"ssm:DescribeAutomationExecutions",
"cloudformation:DescribeStackSet",
"cloudformation:DescribeStackInstance",
"cloudformation:DescribeStacks",
"cloudformation:DescribeStackResources",
"cloudformation:ListStackSetOperations",
"cloudformation:ListStackSets",
"cloudformation:ListStacks",
"cloudformation:ListStackInstances",
"cloudformation:ListStackSetOperationResults",
"cloudformation:TagResource",
"cloudformation:CreateStack",
"cloudformation:DeleteStackSet",
"cloudformation:UpdateStackSet",
"cloudformation:CreateStackSet",
"cloudformation:DeleteStackInstances",
"cloudformation:CreateStackInstances"
```

Management account

```
"ssm:createResourceDataSync",
"ssm:listResourceDataSync",
"ssm:getOpsSummary",
"ssm:createAssociation",
"ssm:createDocument",
"ssm:startAssociationsOnce",
"ssm:startAutomationExecution",
"ssm:updateAssociation",
"ssm:listAssociations",
```

```
"ssm:listDocuments",
"ssm:getDocument",
"ssm:describeAssociation",
"ssm:describeAutomationExecutions",
"organizations:ListRoots",
"organizations:DescribeOrganization",
"organizations:ListOrganizationalUnitsForParent"
"organizations:EnableAWSServiceAccess",
"cloudformation:describe*"
```

Utiliser Quick Setup

Quick Setup, une fonctionnalité de AWS Systems Manager, affiche les résultats de chaque configuration dans le tableau Configurations de la page d'accueil Quick Setup. À partir de cette page, vous pouvez sélectionner View details (Afficher les détails) de chaque configuration, supprimer des configurations à partir de la liste déroulante Actions ou sélectionner Create (Créer) pour en créer. Le tableau Configurations contient les informations suivantes :

- Configuration type (Type de configuration) – Type de configuration choisi lors de la création de la configuration.
- Type de déploiement : indique si le déploiement s'applique à l'ensemble de l'organisation (`Organizational`) ou uniquement à votre compte (`Local`).
- Organizational units (Unités organisationnelles) – Affiche les unités organisationnelles (UO) vers lesquelles la configuration est déployée si vous avez choisi un ensemble de cibles Custom (Personnalisé). Les unités d'organisation et les cibles personnalisées ne sont disponibles que pour le compte principal de votre organisation. Le compte de gestion est le compte que vous utilisez pour créer une organisation dans AWS Organizations.
- Regions (Régions) – Régions vers lesquelles la configuration est déployée si vous avez choisi un ensemble de cibles Custom (Personnalisé) ou des cibles au sein de votre Current account (Compte actuel).
- Deployment status (Statut du déploiement) – Le statut du déploiement indique si AWS CloudFormation a réussi à déployer l'instance cible ou de pile. Les instances cible et de pile contiennent les options de configuration que vous avez choisies lors de la création de la configuration.

- Association status (Statut d'association) – Le statut de l'association est l'état de toutes les associations créées par la configuration que vous avez créée. Les associations pour toutes les cibles doivent s'exécuter correctement. Sinon, le statut est Failed (Échec).

  Quick Setup crée et exécute une association State Manager pour chaque cible de configuration. State Manager est une fonctionnalité de AWS Systems Manager.

Détails de configuration

La page Configuration details (Détails de la configuration) affiche des informations sur le déploiement de la configuration et de ses associations. À partir de cette page, vous pouvez modifier les options de configuration, mettre à jour des cibles ou supprimer la configuration. Vous pouvez également afficher les détails de chaque déploiement de configuration pour obtenir plus d'informations sur les associations.

Selon le type de configuration, un ou plusieurs des graphiques d'état suivants s'affichent :

Statut du déploiement de configuration

Affiche le nombre de déploiements qui ont réussi, échoué, sont en cours d'exécution ou en attente. Les déploiements se produisent dans les comptes et régions des cibles spécifiés qui contiennent des nœuds concernés par la configuration.

Statut de l'association de configuration

Affiche le nombre d'associations State Manager qui ont réussi, échoué ou sont en attente. Quick Setup crée une association dans chaque déploiement pour les options de configuration sélectionnées.

Statut de la configuration

Affiche le nombre d'actions effectuées par type de configuration et leur statut actuel.

Conformité des ressources

Affiche le nombre de ressources qui sont conformes à la politique de configuration spécifiée.

Le tableau Configuration details (Détails de la configuration) affiche les informations sur le déploiement de votre configuration. Vous pouvez afficher plus de détails sur chaque déploiement en sélectionnant le déploiement, puis l'option View details (Afficher les détails). La page des détails de chaque déploiement affiche les associations déployées sur les nœuds de ce déploiement.

Modification et suppression de votre configuration

Vous pouvez modifier les options de configuration d'une configuration à partir de la page Configuration details (Détails de la configuration) en sélectionnant Actions, puis Edit configuration options (Modifier les options de configuration). Lorsque vous ajoutez de nouvelles options à la configuration, Quick Setup exécute vos déploiements et crée d'autres associations. Lorsque vous supprimez des options d'une configuration, Quick Setup exécute vos déploiements et supprime toutes les associations liées.

> **Note**
>
> Vous pouvez modifier les configurations Quick Setup à tout moment pour votre compte. Pour modifier la configuration Organization, le Configuration status (Statut de la configuration) doit être Success (Succès) ou Failed (Échec).

Vous pouvez également mettre à jour les cibles incluses dans vos configurations en sélectionnant Actions et Add OUs (Ajouter des UO), Add Regions (Ajouter des régions), Remove OUs (Supprimer des UO) ou Remove Regions (Supprimer des régions). Si votre compte n'est pas configuré en tant que compte de gestion ou si vous avez créé la configuration uniquement pour le compte courant, vous ne pouvez pas mettre à jour les unités d'organisation cibles (UO). La suppression d'une région ou d'une unité organisationnelle supprime les associations de ces régions ou unités organisationnelles.

Vous pouvez supprimer une configuration de Quick Setup en sélectionnant la configuration, puis Actions, puis Delete configuration (Supprimer la configuration). Ou vous pouvez supprimer la configuration à partir de la page Configuration details (Détails de la configuration) depuis la liste déroulante Actions (Actions), puis choisir Delete configuration (Supprimer la configuration). Quick Setup vous invitera ensuite à Remove all OUs and Regions (Supprimer toutes les UO et régions). Cette opération peut prendre un peu de temps. La suppression d'une configuration supprime également toutes les associations liées. Ce processus de suppression en deux étapes supprime toutes les ressources déployées de tous les comptes et régions, puis supprime la configuration.

Conformité de la configuration

Vous pouvez voir si vos instances sont conformes aux associations créées par vos configurations dans Explorer ou Conformité, tous deux des fonctionnalités de AWS Systems Manager. Pour en

savoir plus sur la conformité, consultez Utilisation du service Conformité. Pour en savoir plus sur l'affichage de la conformité Explorer, consultez AWS Systems Manager Explorer.

Types de configuration Quick Setup pris en charge

Types de configuration pris en charge

Quick Setup prend en charge les types de configuration suivants.

- Gestion des hôtes Amazon EC2
- Gestion des hôtes par défaut pour une organisation
- Enregistreur de configuration AWS Config
- AWS Config déploiement du pack de conformité
- Configuration des correctifs de l'organisation Patch Manager
- Configuration de l'organisation Change Manager
- DevOpsConfiguration du gourou
- Déploiement du package Distributor
- Planification des ressources de l'instance Amazon EC2
- Configuration de l'organisation OpsCenter
- Explorateur de ressources AWS configuration

Gestion des hôtes Amazon EC2

Utilisez Quick Setup une fonctionnalité permettant de AWS Systems Manager configurer rapidement les rôles de sécurité requis et les fonctionnalités de Systems Manager couramment utilisées sur vos instances Amazon Elastic Compute Cloud (Amazon EC2). Vous pouvez l'utiliser Quick Setup dans un compte individuel ou sur plusieurs comptes et Régions AWS en 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.

Si les services et fonctions Systems Manager sont nouveaux pour vous, nous vous recommandons d'étudier le Guide de l'utilisateur AWS Systems Manager avant de créer une configuration avec Quick Setup. Pour de plus amples informations sur Systems Manager, veuillez consulter Qu'est-ce que c'est AWS Systems Manager ?.

Important

Quick Setup n'est peut-être pas l'outil approprié à utiliser pour la gestion d'EC2 si l'une des conditions suivantes vous concerne :

- Vous essayez de créer une instance EC2 pour la première fois afin de tester les AWS fonctionnalités.
- Vous débutez dans la gestion des instances EC2.

Nous vous recommandons plutôt de commencer par la configuration suivante :

- Mise en route avec Amazon EC2
- Lancez une instance à l'aide du nouvel assistant de lancement d'instance du guide de l'utilisateur Amazon EC2
- Lancez une instance à l'aide du nouvel assistant de lancement d'instance du guide de l'utilisateur Amazon EC2
- Tutoriel : Commencez à utiliser les instances Linux Amazon EC2 dans le guide de l'utilisateur Amazon EC2

Si vous maitrisez déjà la gestion des instances EC2 et que vous souhaitez rationaliser la configuration ainsi que la gestion de plusieurs instances EC2, utilisez Quick Setup. Que votre entreprise possède des dizaines, des milliers ou des millions d'instances EC2, utilisez la procédure Quick Setup suivante pour configurer plusieurs options pour ces instances, en une seule fois.

Prérequis

La région d'origine pour Quick Setup doit déjà être spécifiée avant de terminer les tâches suivantes. Pour plus d'informations, veuillez consulter Configuration de la Région AWS d'accueil.

Note

Ce type de configuration vous permet de définir plusieurs options pour l'ensemble d'une organisation définie dans AWS Organizations, uniquement certains comptes d'organisation et certaines régions, ou pour un seul compte. L'une de ces options consiste à vérifier et à appliquer les mises à jour pour SSM Agent toutes les deux semaines. Si vous êtes

administrateur d'une organisation, vous pouvez également choisir de mettre à jour toutes les instances EC2 de votre organisation avec des mises à jour de l'agent toutes les deux semaines en utilisant le type de Configuration de gestion des hôtes par défaut. Pour plus d'informations, veuillez consulter Gestion des hôtes par défaut pour une organisation.

Configuration des options de gestion de l'hôte pour les instances EC2

Pour configurer la gestion des hôtes, effectuez les tâches suivantes dans la AWS Systems Manager Quick Setup console.

Pour ouvrir la page de configuration de la gestion des hôtes

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Quick Setup.
3. Sur la carte Gestion des hôtes, sélectionnez Créer.

> **Tip**
>
> Si vous avez déjà une ou plusieurs configurations dans votre compte, choisissez d'abord l'onglet Bibliothèque ou le bouton Créer dans la section Configurations pour afficher les cartes.

Pour configurer les options de gestion des hôtes de Systems Manager

- Pour configurer les fonctionnalités de Systems Manager, dans la section Options de configuration, choisissez les options du groupe Systems Manager que vous souhaitez activer pour votre configuration :

  Mettez à jour l'agent Systems Manager (SSM) toutes les deux semaines

  Permet à Systems Manager de vérifier toutes les deux semaines la présence d'une nouvelle version de l'agent. S'il existe une nouvelle version, Systems Manager met automatiquement à jour l'agent sur votre nœud géré vers la dernière version publiée. Quick Setup n'installe pas l'agent sur les instances où il n'est pas déjà présent. Pour plus d'informations sur quelles

AMIs ont SSM Agent de préinstallés, consultez Rechercher AMIs avec le SSM Agent préinstallé.

Nous vous encourageons à choisir cette option pour vous assurer que vos nœuds exécutent toujours la up-to-date version la plus complète deSSM Agent. Pour de plus amples informations sur SSM Agent, y compris sur la façon d'installer manuellement l'agent, veuillez consulter Utilisation de l'option SSM Agent.

Collectez l'inventaire de vos instances toutes les 30 minutes

Permet Quick Setup de configurer la collecte des types de métadonnées suivants :

- AWS composants : pilote EC2, agents, versions, etc.
- Applications – noms des applications, éditeurs, versions, etc.
- Détails du nœud – 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, état, services dépendants, type de service, type de démarrage, etc. (nœuds Windows Server uniquement).
- Rôles Windows – nom, nom d'affichage, chemin, type de fonctionnalité, état installé, etc. (nœuds Windows Server uniquement).
- Mises à jour Windows – ID de correctif, installé par, date d'installation, etc. (nœuds Windows Server uniquement).

Pour de plus amples informations sur Inventory, une fonctionnalité de AWS Systems Manager, veuillez consulter AWS Systems Manager Inventory.

> **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 nœuds.

Analyser quotidiennement les instances pour rechercher les correctifs manquants

PermetPatch Manager, grâce à une fonctionnalité de Systems Manager, de scanner vos nœuds quotidiennement et de générer un rapport sur la page Conformité. Le rapport indique

combien de nœuds sont conformes aux correctifs en fonction du référentiel de correctifs par défaut. Le rapport inclut une liste de chaque nœud et de son statut de conformité.

Pour obtenir des informations sur les opérations d'application de correctifs et les référentiels de correctifs, veuillez consulter la rubrique AWS Systems Manager Patch Manager.

Pour obtenir des informations sur la conformité aux correctifs, veuillez consulter la page Compliance (Conformité) de Systems Manager.

Pour obtenir des informations sur l'application de correctifs à des nœuds gérés sur plusieurs comptes et dans plusieurs régions dans une seule configuration, veuillez consulter les rubriques Utilisation des stratégies de correctifs Quick Setup et Configuration des correctifs de l'organisation Patch Manager.

> **Important**
>
> Systems Manager prend en charge plusieurs méthodes d'analyse de conformité aux correctifs des nœuds gérés. Si vous implémentez plusieurs de ces méthodes à la fois, les informations de conformité aux correctifs que vous voyez sont toujours le résultat de l'analyse la plus récente. Les résultats des analyses précédentes sont remplacés. Si les méthodes d'analyse utilisent des référentiels de correctifs différents, avec des règles d'approbation différentes, les informations de conformité aux correctifs peuvent changer de manière inattendue. Pour plus d'informations, consultez Éviter les remplacements involontaires des données de conformité aux correctifs.

Pour configurer les options de gestion des CloudWatch hôtes Amazon

- Pour configurer les CloudWatch fonctionnalités, dans la section Options de configuration, choisissez les options du CloudWatch groupe Amazon que vous souhaitez activer pour votre configuration :

  Installation et configuration de l' CloudWatch agent

  Installe la configuration de base de l' CloudWatch agent unifié sur vos instances Amazon EC2. L'agent collecte des métriques et des fichiers journaux à partir de vos instances pour

Amazon CloudWatch. Ces informations sont consolidées pour vous permettre de déterminer rapidement l'état de vos instances. Pour plus d'informations sur la configuration de base de l' CloudWatch agent, consultez les ensembles de mesures prédéfinis par l'CloudWatch agent. Des frais supplémentaires pourraient vous être facturés. Pour plus d'informations, consultez les CloudWatchtarifs Amazon.

Mettez à jour l' CloudWatch agent une fois tous les 30 jours

Permet à Systems Manager de vérifier tous les 30 jours la présence d'une nouvelle version de l' CloudWatch agent. S'il existe une nouvelle version, Systems Manager met à jour l'agent sur votre instance. Nous vous encourageons à choisir cette option pour vous assurer que vos instances exécutent toujours la up-to-date version la plus complète de l' CloudWatch agent.

Pour configurer les options de gestion des hôtes de l'agent de lancement Amazon EC2

- Pour configurer les fonctionnalités de l'agent de lancement Amazon EC2, dans la section Options de configuration, choisissez les options du groupe d'agents de lancement Amazon EC2 que vous souhaitez activer pour votre configuration :

Mettez à jour l'agent de lancement EC2 une fois tous les 30 jours

Permet à Systems Manager de vérifier tous les 30 jours si une nouvelle version de l'agent de lancement est installée sur votre instance. Si une nouvelle version est disponible, Systems Manager met à jour l'agent sur votre instance. Nous vous encourageons à choisir cette option pour vous assurer que vos instances exécutent toujours la up-to-date version la plus complète de l'agent de lancement applicable. Pour les instances Amazon EC2 Windows, cette option prend en charge EC2Launch, EC2Launch v2 et EC2Config. Pour les instances Linux Amazon EC2, cette option prend en charge `cloud-init`. Pour les instances Amazon EC2 Mac, cette option prend en charge `ec2-macos-init`. Quick Setup ne prend pas en charge la mise à jour des agents de lancement installés sur des systèmes d'exploitation non pris en charge par l'agent de lancement ou sur AL2023.

Pour plus d'informations sur ces agents d'initialisation, consultez les rubriques suivantes :

- Configurer une instance Windows à l'aide d'EC2Launch v2
- Configurer une instance Windows à l'aide d'EC2Launch
- Configurer une instance Windows à l'aide du service EC2Config

- Documentation cloud-init
- ec2-macos-init

Pour sélectionner les instances EC2 à mettre à jour par la configuration de gestion de l'hôte

- Dans la section Cibles, choisissez la méthode pour déterminer les comptes et les régions dans lesquels la configuration doit être déployée :

> **Note**
>
> Vous ne pouvez pas créer plusieurs configurations de gestion des hôtes Quick Setup ciblant la même Région AWS.

Entire organization

Votre configuration est déployée dans toutes les unités organisationnelles (UO) et Régions AWS au sein de votre organisation.

> **Note**
>
> L'option Ensemble de l'organisation est disponible uniquement si vous configurez la gestion des hôtes à partir du compte de gestion de votre organisation.

Custom

1. Dans la section UO cibles, sélectionnez les unités d'organisation dans lesquelles vous souhaitez déployer cette configuration de gestion d'hôte.
2. Dans la section Régions cibles, sélectionnez les régions dans lesquelles vous souhaitez déployer cette configuration de gestion d'hôte.

Current account

Choisissez l'une des options de région et suivez les étapes associées à cette option.

Région actuelle

Choisissez le mode de ciblage des instances de la région actuelle uniquement :

- Toutes les instances : la configuration de gestion des hôtes cible automatiquement tous les EC2 de la région actuelle.
- Tag — Choisissez Ajouter et entrez la clé et la valeur facultative qui sont ajoutées aux instances à cibler.
- Groupe de ressources : pour Groupe de ressources, sélectionnez un groupe de ressources existant qui contient les instances EC2 à cibler.
- Manuel — Dans la section Instances, cochez la case de chaque instance EC2 à cibler.

Choisissez les régions

Choisissez comment cibler les instances dans la région que vous spécifiez en choisissant l'une des options suivantes :

- Toutes les instances : toutes les instances des régions que vous spécifiez sont ciblées.
- Tag — Choisissez Ajouter et entrez la clé et la valeur facultative qui ont été ajoutées aux instances à cibler.

Dans la section Régions cibles, sélectionnez les régions dans lesquelles vous souhaitez déployer cette configuration de gestion d'hôte.

Pour spécifier une option de profil d'instance

- Organisation entière et cibles personnalisées uniquement.

  Dans la section Options du profil d'instance, choisissez si vous souhaitez ajouter les politiques IAM requises aux profils d'instance existants attachés à vos instances, ou autoriser Quick Setup la création des politiques IAM et des profils d'instance avec les autorisations nécessaires pour la configuration que vous choisissez.

Après avoir défini tous vos choix de configuration, choisissez Create.

Gestion des hôtes par défaut pour une organisation

Grâce à Quick Setup cette fonctionnalité AWS Systems Manager, vous pouvez activer la configuration de gestion d'hôte par défaut pour tous les comptes et régions ajoutés à votre

organisation dans AWS Organizations. Cela garantit que SSM Agent est tenu à jour sur toutes les instances Amazon Elastic Compute Cloud (EC2) de l'organisation et qu'elles peuvent se connecter à Systems Manager.

Avant de commencer

Vérifiez que les conditions suivantes sont respectées avant d'activer ce paramètre.

- La région d'origine pour Quick Setup doit déjà être spécifiée avant de terminer les tâches suivantes. Pour plus d'informations, veuillez consulter Configuration de la Région AWS d'accueil.
- La dernière version de SSM Agent est déjà installée sur toutes les instances EC2 à gérer dans votre organisation.
- Votre instances EC2 à gérer utilisent le service de métadonnées d'instance version 2 (IMDSv2).
- Vous êtes connecté au compte de gestion de votre organisation, comme indiqué dans AWS Organizations, à l'aide d'une identité AWS Identity and Access Management (IAM) (utilisateur, rôle ou groupe) avec des autorisations d'administrateur.

Utilisation du rôle de gestion d'instance EC2 par défaut

La Configuration de gestion des hôtes par défaut utilise le paramètre de service `default-ec2-instance-management-role` pour Systems Manager. Il s'agit d'un rôle doté d'autorisations que vous souhaitez mettre à la disposition de tous les comptes de votre organisation afin de permettre la communication entre les SSM Agent sur l'instance et le service Systems Manager dans le cloud.

Si vous avez déjà défini ce rôle à l'aide de la commande CLI update-service-setting, la Configuration de gestion des hôtes par défaut utilise ce rôle. Si vous n'avez pas encore défini ce rôle, Quick Setup le créera et l'appliquera pour vous.

Pour vérifier si ce rôle a déjà été spécifié pour votre organisation, utilisez la commande get-service-setting.

Activer les mises à jour automatiques de SSM Agent toutes les deux semaines

Utilisez la procédure suivante pour activer l'option de configuration de gestion d'hôte par défaut pour AWS Organizations l'ensemble de votre organisation.

Pour activer les mises à jour automatiques de SSM Agent toutes les deux semaines

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.

2. Dans le panneau de navigation, sélectionnez Quick Setup.
3. Sur la carte Configuration de gestion des hôtes par défaut, sélectionnez Créer.

> **Tip**
>
> Si vous avez déjà une ou plusieurs configurations dans votre compte, choisissez d'abord l'onglet Bibliothèque ou le bouton Créer dans la section Configurations pour afficher les cartes.

4. Dans la section Options de configuration, sélectionnez Activer les mises à jour automatiques de SSM Agent toutes les deux semaines.
5. Sélectionnez Create (Créer).

Enregistreur de configuration AWS Config

Avec Quick Setup une fonctionnalité de AWS Systems Manager, vous pouvez créer rapidement un enregistreur de configuration alimenté par AWS Config. Utilisez l'enregistreur de configuration pour détecter les changements apportés à vos configurations de ressources et capturer les changements en tant qu'éléments de configuration. Si vous ne le connaissez pas AWS Config, nous vous recommandons d'en savoir plus sur le service en consultant le contenu du guide du AWS Config développeur avant de créer une configuration avecQuick Setup. Pour plus d'informations AWS Config, voir Qu'est-ce que c'est AWS Config ? dans le Guide AWS Config du développeur.

Par défaut, l'enregistreur de configuration enregistre toutes les ressources prises en charge dans Région AWS le AWS Config répertoire d'exécution. Vous pouvez personnaliser la configuration de sorte que seuls les types de ressources que vous spécifiez soient enregistrés. Pour plus d'informations, consultez la section Sélection des ressources AWS Config enregistrées dans le Guide du AWS Config développeur.

Des frais d'utilisation du service vous sont facturés lorsque vous AWS Config commencez à enregistrer des configurations. Pour en savoir plus sur la tarification, consultez Tarification AWS Config.

> **Note**
>
> Si vous avez déjà créé un enregistreur de configuration, Quick Setup n'arrête pas l'enregistrement et n'apporte aucune modification aux types de ressources que vous enregistrez déjà. Si vous choisissez d'enregistrer des types de ressources supplémentaires

à l'aide de Quick Setup, le service les ajoute à vos groupes d'enregistreurs existants. La suppression du type de configuration de l'Enregistrement de config Quick Setup n'arrête pas l'enregistreur de configuration. Les modifications continuent d'être enregistrées et des frais d'utilisation du service s'appliquent jusqu'à ce que vous arrêtiez l'enregistreur de configuration. Pour en savoir plus sur la gestion de l'enregistreur de configuration, veuillez consulter Gestion de l'enregistreur de configuration dans le Manuel du développeur AWS Config .

Prérequis

La région d'origine pour Quick Setup doit déjà être spécifiée avant de terminer les tâches suivantes. Pour plus d'informations, veuillez consulter Configuration de la Région AWS d'accueil.

Pour configurer AWS Config l'enregistrement, effectuez les tâches suivantes dans la AWS Systems Manager console.

Pour configurer AWS Config l'enregistrement avec Quick Setup

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Quick Setup.
3. Sur la carte Enregistrement de configuration, sélectionnez Créer.

   > **Tip**
   >
   > Si vous avez déjà une ou plusieurs configurations dans votre compte, choisissez d'abord l'onglet Bibliothèque ou le bouton Créer dans la section Configurations pour afficher les cartes.

4. Dans la section Options de configuration, procédez comme suit :

   a. Pour Choisir les types de AWS ressources à enregistrer, indiquez si vous souhaitez enregistrer toutes les ressources prises en charge ou uniquement les types de ressources que vous choisissez.

   b. Pour les paramètres de livraison, indiquez si vous souhaitez créer un nouveau compartiment Amazon Simple Storage Service (Amazon S3) ou choisir un compartiment existant auquel envoyer des instantanés de configuration.

c. Pour les options de notification, choisissez l'option de notification que vous préférez. AWS Config utilise Amazon Simple Notification Service (Amazon SNS) pour vous informer des événements AWS Config importants liés à vos ressources. Si vous choisissez l'option Utiliser les rubriques SNS existantes, vous devez fournir l' Compte AWS ID et le nom de la rubrique Amazon SNS existante dans le compte que vous souhaitez utiliser. Si vous ciblez plusieurs Régions AWS, les noms des rubriques doivent être identiques dans chaque Région.

5. Dans la section Planification, sélectionnez la fréquence à laquelle vous voulez que Quick Setup corrige les modifications apportées aux ressources qui diffèrent de votre configuration. L'option Défaut (Par défaut) s'exécute une seule fois. Si vous ne voulez pas que Quick Setup corrige les modifications apportées aux ressources qui diffèrent de votre configuration, sélectionnez Disable remediation (Désactiver la correction) sous Personnalisé.
6. Dans la section Cibles, choisissez l'une des options suivantes pour identifier les comptes et les régions à enregistrer.

> **Note**
>
> Si vous utilisez un seul compte, les options permettant de travailler avec des organisations et des unités organisationnelles (UO) ne sont pas disponibles. Vous pouvez choisir d'appliquer cette configuration à l'ensemble Régions AWS de votre compte ou uniquement aux régions que vous sélectionnez.

- Entire organization (Organisation entière) : tous les comptes et toutes les régions de votre organisation.
- Personnalisé : uniquement les UO et les régions que vous spécifiez.
  - Dans la section UO cibles, sélectionnez les UO pour lesquelles vous souhaitez autoriser l'enregistrement.
  - Dans la section Régions cibles, sélectionnez les régions dans lesquelles vous souhaitez autoriser l'enregistrement.
- Current account (Compte actuel) : seules les régions que vous spécifiez dans le compte auquel vous êtes actuellement connecté sont ciblées. Sélectionnez l'une des méthodes suivantes :
  - Current Region (Région actuelle) : seuls les nœuds gérés de la région sélectionnée dans la console sont ciblés.

- Choisir les régions — Choisissez les régions individuelles auxquelles appliquer la configuration d'enregistrement.

7. Sélectionnez Create (Créer).

AWS Config déploiement du pack de conformité

Un pack de conformité est un ensemble de AWS Config règles et d'actions correctives. Avec Quick Setup, vous pouvez déployer un pack de conformité en tant qu'entité unique dans un compte et une Région AWS , ou dans une organisation, dans AWS Organizations. Cela vous permet de gérer la conformité de la configuration de vos AWS ressources à grande échelle, de la définition des politiques à l'audit et aux rapports agrégés, en utilisant un cadre et un modèle de packaging communs.

Prérequis

La région d'origine pour Quick Setup doit déjà être spécifiée avant de terminer les tâches suivantes. Pour plus d'informations, veuillez consulter Configuration de la Région AWS d'accueil.

Pour déployer des packs de conformité, effectuez les tâches suivantes dans la AWS Systems Manager Quick Setup console.

> **Note**
>
> Vous devez activer AWS Config l'enregistrement avant de déployer cette configuration. Pour plus d'informations, consultez Packs de conformité dans le Guide du développeur AWS Config .

Pour déployer des packs de conformité avec Quick Setup

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Quick Setup.
3. Sur la carte Packs de conformité, sélectionnez Créer.

> **Tip**
>
> Si vous avez déjà une ou plusieurs configurations dans votre compte, choisissez d'abord l'onglet Bibliothèque ou le bouton Créer dans la section Configurations pour afficher les cartes.

4. Dans la section Choisir les packs de conformité, sélectionnez les packs de conformité que vous voulez déployer.

> **Note**
>
> Outre les packs de conformité AWS gérés, vous pouvez choisir parmi les packs de conformité personnalisés que vous avez créés. Pour plus d'informations, consultez les rubriques suivantes du guide du AWS Config développeur :
>
> - Packs de conformité personnalisés
> - Déploiement d'un pack de conformité à l'aide de la console AWS Config
> - Déploiement d'un pack de conformité à l'aide du AWS Command Line Interface

5. Dans la section Planification, sélectionnez la fréquence à laquelle vous voulez que Quick Setup corrige les modifications apportées aux ressources qui diffèrent de votre configuration. L'option Défaut (Par défaut) s'exécute une seule fois. Si vous ne voulez pas que Quick Setup corrige les modifications apportées aux ressources qui diffèrent de votre configuration, sélectionnez Disabled (Désactivé) sous Custom (Personnalisé).
6. Dans la section Cibles, choisissez de déployer des packs de conformité sur l'ensemble de votre organisation, sur certaines Régions AWS d'entre elles ou sur le compte auquel vous êtes actuellement connecté.

   Si vous sélectionnez Entire organization (Ensemble de l'organisation), passez à l'étape 8.

   Si vous sélectionnez Personnalisé, passez à l'étape 7.
7. Dans la section Target Regions (Régions cibles), cochez les cases des régions dans lesquelles vous voulez déployer des packs de conformité.
8. Sélectionnez Create (Créer).

Configuration des correctifs de l'organisation Patch Manager

Avec Quick Setup une fonctionnalité de AWS Systems Manager, vous pouvez créer des politiques de correctif basées surPatch Manager. Une politique de correctifs définit la planification et le référentiel à utiliser lorsque des correctifs sont automatiquement appliqués à vos instances Amazon Elastic Compute Cloud (Amazon EC2) et à d'autres nœuds gérés. À l'aide d'une configuration de politique de correctifs unique, vous pouvez définir l'application de correctifs pour tous les comptes de plusieurs Régions AWS de votre organisation, uniquement pour les comptes et les régions de votre choix, ou pour une seule paire compte-région. Pour plus d'informations sur les politiques de correctifs, consultez la rubrique Utilisation des stratégies de correctifs Quick Setup.

Prérequis

Pour définir une politique de correctifs pour un nœud avec Quick Setup, le nœud doit être un nœud géré. Pour plus d'informations sur la gestion de vos nœuds, consultez la rubrique Con AWS Systems Manager figuration.

> **Important**
>
> Méthodes d'analyse de conformité des correctifs : Systems Manager prend en charge plusieurs méthodes pour analyser les nœuds gérés afin de vérifier la conformité des correctifs. Si vous implémentez plusieurs de ces méthodes à la fois, les informations de conformité aux correctifs que vous voyez sont toujours le résultat de l'analyse la plus récente. Les résultats des analyses précédentes sont remplacés. Si les méthodes d'analyse utilisent des référentiels de correctifs différents, avec des règles d'approbation différentes, les informations de conformité aux correctifs peuvent changer de manière inattendue. Pour plus d'informations, consultez Éviter les remplacements involontaires des données de conformité aux correctifs.
>
> État de conformité des associations et politiques relatives aux correctifs : l'état des correctifs pour un nœud géré soumis à une politique de Quick Setup correctifs correspond au statut de l'exécution de l'State Managerassociation pour ce nœud. Si l'état d'exécution de l'association est`Compliant`, l'état d'application des correctifs pour le nœud géré est également marqué`Compliant`. Si l'état d'exécution de l'association est`Non-Compliant`, l'état d'application des correctifs pour le nœud géré est également marqué`Non-Compliant`.

Régions prises en charge pour les configurations des politiques de correctif

Actuellement, les configurations d'application de correctifs de Quick Setup sont prises en charge dans les régions suivantes :

- USA Est (Ohio) (us-east-2)
- USA Est (Virginie du Nord) (us-east-1)
- US Ouest (N. California) (us-west-1)
- USA Ouest (Oregon) (us-west-2)
- Asie-Pacifique (Mumbai) (ap-south-1)
- 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)

Autorisations pour le compartiment S3 de la politique de correctifs

Lorsque vous créez une politique de correctifs, Quick Setup crée un compartiment Amazon S3 qui contient un fichier nommé `baseline_overrides.json`. Ce fichier contient des informations sur les référentiels de correctifs que vous avez spécifiées pour votre politique de correctifs.

Le compartiment S3 est nommé au format `aws-quicksetup-patchpolicy-`*`account-id`*`-`*`quick-setup-configuration-id`*.

Par exemple : `aws-quicksetup-patchpolicy-123456789012-abcde`

Si vous créez une politique de correctifs pour une organisation, le compartiment est créé dans le compte de gestion de votre organisation.

Il existe deux cas d'utilisation dans lesquels vous devez autoriser d'autres AWS ressources à accéder à ce compartiment S3 à l'aide de politiques AWS Identity and Access Management (IAM) :

- Cas 1 : utiliser votre propre profil d'instance ou fonction du service avec vos nœuds gérés au lieu d'un profil fourni par Quick Setup
- Cas 2 : utiliser les points de terminaison d'un VPC pour se connecter à Systems Manager

La politique d'autorisations dont vous avez besoin dans les deux cas se trouve dans la section ci-dessous, Autorisations des politiques pour les compartiments S3 Quick Setup.

Cas 1 : utiliser votre propre profil d'instance ou fonction du service avec vos nœuds gérés au lieu d'un profil fourni par Quick Setup

Les configurations des politiques de correctifs comprennent une option pour Ajouter les politiques IAM requises aux profils d'instance existants attachés à vos instances.

Si vous ne choisissez pas cette option, mais que vous souhaitez que Quick Setup applique des correctifs à vos nœuds gérés à l'aide de cette politique de correctif, vous devez vous assurer que les éléments suivants sont mis en œuvre :

- La politique `AmazonSSMManagedInstanceCore` gérée par IAM doit être associée au profil d'instance IAM ou à la fonction du service IAM utilisé pour fournir des autorisations Systems Manager à vos nœuds gérés.
- Vous devez ajouter au profil d'instance IAM ou à la fonction du service IAM des autorisations d'accès à votre compartiment de politiques de correctifs en tant que politique en ligne. Vous pouvez fournir un accès générique à tous les compartiments `aws-quicksetup-patchpolicy` ou uniquement au compartiment spécifique créé pour votre organisation ou votre compte, comme indiqué dans les exemples de code précédents.
- Vous devez baliser votre profil d'instance IAM ou votre fonction du service IAM à l'aide de la paire clé-valeur suivante.

  `Key: QSConfigId-`*`quick-setup-configuration-id`*`, Value: `*`quick-setup-configuration-id`*

  `quick-setup-configuration-id` représente la valeur du paramètre appliqué à la AWS CloudFormation pile utilisée pour créer votre configuration de politique de correctifs. Pour récupérer cet ID, procédez comme suit :

1. Ouvrez la AWS CloudFormation console à l'adresse https://console.aws.amazon.com/cloudformation.
2. Sélectionnez le nom de la pile utilisée pour créer votre stratégie de correctif. Le nom est dans un format tel que `StackSet-AWS-QuickSetup-PatchPolicy-LA-q4bkg-52cd2f06-d0f9-499e-9818-d887cEXAMPLE`.
3. Sélectionnez l'onglet Paramètres.
4. Dans la liste des paramètres, dans la colonne Clé, recherchez la clé QS ConfigurationId. Dans la colonne Valeur de sa ligne, recherchez l'ID de configuration, tel que `abcde`.

   Dans cet exemple, pour que la balise s'applique à votre profil d'instance ou à votre fonction du service, la clé est `QSConfigId-abcde` et la valeur est `abcde`.

Pour plus d'informations sur l'ajout de balises à un rôle IAM, consultez les sections Balisage des rôles IAM et Gestion des balises sur les profils d'instance (AWS CLI ou AWS API) dans le guide de l'utilisateur IAM.

Cas 2 : utiliser les points de terminaison d'un VPC pour se connecter à Systems Manager

Si vous utilisez des points de terminaison d'un VPC pour vous connecter à Systems Manager, votre politique de point de terminaison d'un VPC pour S3 doit autorise l'accès au compartiment S3 de votre politique de correctifs Quick Setup.

Pour plus d'informations sur l'ajout d'autorisations à une politique de point de terminaison d'un VPC pour S3, consultez la section Contrôle de l'accès depuis les points de terminaison d'un VPC avec des politiques de compartiment dans le Guide de l'utilisateur Amazon S3.

Autorisations des politiques pour les compartiments S3 Quick Setup

Vous pouvez fournir un accès générique à tous les compartiments `aws-quicksetup-patchpolicy` ou uniquement au compartiment spécifique créé pour votre organisation ou votre compte. Pour fournir les autorisations nécessaires dans les deux cas décrits ci-dessous, utilisez l'un ou l'autre format.

All patch policy buckets

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

```
      "Sid": "AccessToAllPatchPolicyRelatedBuckets",
      "Effect": "Allow",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::aws-quicksetup-patchpolicy-*"
    }
  ]
}
```

Specific patch policy bucket

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AccessToMyPatchPolicyRelatedBucket",
      "Effect": "Allow",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::aws-quicksetup-patchpolicy-account-id-quick-setup-configuration-
id"[1]
    }
  ]
}
```

[1]Une fois la configuration de la politique de correctifs créée, vous pouvez localiser le nom complet de votre compartiment dans la console S3. Par exemple : `aws-quicksetup-patchpolicy-123456789012-abcde`

ID référentiel de correctifs aléatoires dans les opérations relatives aux politique de correctifs

Les opérations d'application de correctifs pour les politiques de correctifs utilisent le paramètre `BaselineOverride` figurant dans le document de commande SSM `AWS-RunPatchBaseline`.

Lorsque vous l'utilisez `AWS-RunPatchBaseline` pour appliquer des correctifs en dehors d'une politique de correctifs, vous pouvez utiliser `BaselineOverride` pour spécifier une liste de référentiels de correctifs à utiliser pendant l'opération qui sont différentes des valeurs par défaut spécifiées. Vous créez cette liste dans un fichier nommé `baseline_overrides.json` et vous l'ajoutez manuellement à un compartiment Amazon S3 que vous possédez, comme expliqué dans Utilisation du BaselineOverride paramètre.

Toutefois, pour les opérations d'application de correctifs basées sur des politiques de correctifs, Systems Manager crée automatiquement un compartiment S3 et y ajoute un fichier `baseline_overrides.json`. Ensuite, chaque fois que Quick Setup exécute une opération d'application de correctifs (à l'aide de la fonctionnalité Run Command), le système génère un ID aléatoire pour chaque référentiel de correctifs. Cet ID est différent pour chaque opération d'application de correctifs de la politique de correctifs et le référentiel de correctifs qu'il représente n'est ni stocké ni accessible dans votre compte.

Par conséquent, vous ne verrez pas l'ID du référentiel de correctifs sélectionné dans votre configuration dans les journaux d'application de correctifs. Cela s'applique à la fois aux lignes de base de correctifs AWS gérées et aux lignes de base de correctifs personnalisées que vous avez peut-être sélectionnées. L'ID de référence indiqué dans le journal est plutôt celui qui a été généré pour cette opération spécifique d'application de correctifs.

En outre, si vous essayez d'afficher les détails dans Patch Manager concernant un référentiel de correctifs généré avec un ID aléatoire, le système indique que le référentiel de correctifs n'existe pas. Ce comportement est normal et peut être ignoré.

Création d'une politique de correctif

Prérequis

La région d'origine pour Quick Setup doit déjà être spécifiée avant de terminer les tâches suivantes. Pour plus d'informations, veuillez consulter Configuration de la Région AWS d'accueil.

Pour créer une politique de correctifs, exécutez les tâches suivantes dans la console Systems Manager.

Créer une politique de correctifs avec Quick Setup

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.

   Si vous configurez l'application de correctifs pour une organisation, assurez-vous d'être connecté au compte de gestion de l'organisation. Vous ne pouvez pas configurer la politique à l'aide du compte d'administrateur délégué ou d'un compte membre.
2. Dans le panneau de navigation, sélectionnez Quick Setup.
3. Sur la carte Patch Manager (Gestionnaire de correctifs), choisissez Create (Créer).

> **Tip**
>
> Si vous avez déjà une ou plusieurs configurations dans votre compte, choisissez d'abord l'onglet Bibliothèque ou le bouton Créer dans la section Configurations pour afficher les cartes.

4. Pour Configuration name (Nom de configuration), saisissez un nom permettant d'identifier la politique de correctifs.
5. Dans la section Scanning and installation (Analyse et installation), sous Patch operation (Opération de correctif), choisissez si la politique de correctifs va Scan (Analyser) les cibles spécifiées ou Scan and install (Scanner et installer) des correctifs sur les cibles spécifiées.
6. Sous Scanning schedule (Planification d'analyse), choisissez Use recommended defaults (Utiliser les valeurs par défaut recommandées) ou Custom scan schedule (Planification d'analyse personnalisée). La planification d'analyse par défaut analysera vos cibles tous les jours à 1 h 00, UTC.

   - Si vous choisissez Custom scan schedule (Planification d'analyse personnalisée), sélectionnez la Scanning frequency (Fréquence d'analyse).
   - Si vous choisissez Daily (Quotidien), saisissez l'heure, au format UTC, à laquelle vous souhaitez analyser vos cibles.
   - Si vous choisissez Custom CRON Expression (Expression CRON personnalisée), saisissez la planification en tant que CRON expression (Expression CRON). Pour plus d'informations sur le formatage des expressions CRON pour Systems Manager, consultez la rubrique Référence : Expressions Cron et Rate pour Systems Manager.

     Sélectionnez également Wait to scan targets until first CRON interval (Attendre le premier intervalle CRON pour analyser les cibles). Par défaut, Patch Manager analyse immédiatement les nœuds lorsqu'ils deviennent des cibles.
7. Si vous avez choisi Scan and install (Analyser et installer), choisissez la Installation schedule (Planification d'installation) à utiliser lors de l'installation de correctifs sur les cibles spécifiées. Si vous choisissez Use recommended defaults (Utiliser les valeurs par défaut recommandées), Patch Manager installera des correctifs hebdomadaires le dimanche à 2 h 00, UTC.

   - Si vous choisissez Custom install schedule (Planification d'installation personnalisée), sélectionnez la Installation frequency (Fréquence d'installation).

- Si vous choisissez Daily (Quotidien), saisissez l'heure, au format UTC, à laquelle vous souhaitez installer des mises à jour sur vos cibles.
- Si vous choisissez Custom CRON Expression (Expression CRON personnalisée), saisissez la planification en tant que CRON expression (Expression CRON). Pour plus d'informations sur le formatage des expressions CRON pour Systems Manager, consultez la rubrique Référence : Expressions Cron et Rate pour Systems Manager.

  Désactivez également Wait to install updates until first CRON interval (Attendre le premier intervalle CRON pour installer les mises à jour) pour installer immédiatement les mises à jour sur les nœuds lorsqu'ils deviennent des cibles. Par défaut, Patch Manager attend le premier intervalle CRON pour installer les mises à jour.
- Choisissez Reboot if needed (Redémarrer si nécessaire) pour redémarrer les nœuds après l'installation du correctif. Le redémarrage après l'installation est recommandé, mais peut entraîner des problèmes de disponibilité.

8. Dans la section Patch baseline (Référentiel de correctifs), choisissez les référentiels de correctifs à utiliser lors de l'analyse et de la mise à jour de vos cibles.

   Par défaut, Patch Manager utilise les référentiels de correctifs prédéfinis. Pour plus d'informations, consultez À propos des références prédéfinies.

   Si vous choisissez une ligne de base de correctifs personnalisée, modifiez la ligne de base de correctifs sélectionnée pour les systèmes d'exploitation pour lesquels vous ne souhaitez pas utiliser une ligne de base de AWS correctifs prédéfinie.

   Les référentiels de correctifs disponibles dans Quick Setup, que vous utilisiez des référentiels de correctifs AWS prédéfinis ou des référentiels de correctifs personnalisés, sont ceux de la région d'origine que vous avez sélectionnée.

   > **Note**
   >
   > Si vous utilisez des points de terminaison d'un VPC pour vous connecter à Systems Manager, assurez-vous que votre stratégie de point de terminaison d'un VPC pour S3 autorise l'accès à ce compartiment S3. Pour plus d'informations, consultez Autorisations pour le compartiment S3 de la politique de correctifs.

> **Important**
>
> Si vous utilisez une configuration de politique de correctifs dans Quick Setup, les mises à jour que vous apportez aux référentiels de correctifs personnalisés sont synchronisées avec Quick Setup une fois par heure.
> Si un référentiel de correctifs personnalisé référencé dans une politique de correctifs est supprimé, une bannière s'affiche sur la page Quick Setup Configuration details (Détails de configuration) de votre politique de correctifs. La bannière vous informe que la politique de correctifs fait référence à un référentiel de correctifs qui n'existe plus et que les opérations d'application de correctifs suivantes échoueront. Dans ce cas, revenez à la page Quick Setup Configurations, sélectionnez la configuration Patch Manager, puis choisissez Actions, Edit configuration (Modifier la configuration). Le nom du référentiel de correctifs supprimé est surligné et vous devez sélectionner un nouveau référentiel de correctifs pour le système d'exploitation concerné.

9. (Facultatif) Dans la section Patching log storage (Application de correctifs au stockage des journaux), sélectionnez Write output to S3 bucket (Écrire la sortie dans le compartiment S3) pour stocker les journaux des opérations d'application de correctifs dans un compartiment Amazon S3.

   > **Note**
   >
   > Si vous configurez une politique de correctifs pour une organisation, le compte de gestion de votre organisation doit disposer au moins d'autorisations en lecture seule pour ce compartiment. Toutes les unités organisationnelles incluses dans la politique doivent disposer d'un accès en écriture au compartiment. Pour plus d'informations sur l'octroi de l'accès aux compartiments à différents comptes, consultez l'Exemple 2 : propriétaire d'un compartiment accordant à ses utilisateurs des autorisations entre comptes sur un compartiment dans le Guide de l'utilisateur Amazon Simple Storage Service.

10. Choisissez Parcourir S3 pour sélectionner le compartiment dans lequel vous souhaitez stocker les sorties de journaux des correctifs. Le compte de gestion doit avoir un accès en lecture à ce compartiment. Tous les comptes et cibles non liés à la gestion configurés dans la section Targets (Cibles) doivent disposer d'un accès en écriture au compartiment S3 fourni à des fins de journalisation.

11. Dans la section Targets (Cibles), choisissez l'une des options suivantes pour identifier les comptes et les régions concernés par cette opération de politique de correctifs.

> **Note**
>
> Si vous utilisez un seul compte, les options permettant de travailler avec des organisations et des unités organisationnelles (UO) ne sont pas disponibles. Vous pouvez choisir d'appliquer cette configuration à l'ensemble Régions AWS de votre compte ou uniquement aux régions que vous sélectionnez.

- Entire organization (Organisation entière) : tous les comptes et toutes les régions de votre organisation.
- Personnalisé : uniquement les UO et les régions que vous spécifiez.
  - Dans la section Target OUs (UO cibles), sélectionnez les UO dans lesquelles vous souhaitez configurer la politique de correctifs.
  - Dans la section Target Regions (Régions cibles), sélectionnez les régions dans lesquelles vous souhaitez appliquer la politique de correctifs.
- Current account (Compte actuel) : seules les régions que vous spécifiez dans le compte auquel vous êtes actuellement connecté sont ciblées. Sélectionnez l'une des méthodes suivantes :
  - Current Region (Région actuelle) : seuls les nœuds gérés de la région sélectionnée dans la console sont ciblés.
  - Choose Regions (Choisir les régions) : choisissez les régions individuelles auxquelles appliquer la politique de correctifs.

12. Pour Choose how you want to target instances (Choisissez la manière dont vous souhaitez cibler les instances), choisissez l'une des options suivantes pour identifier les nœuds auxquels appliquer des correctifs :

- All managed nodes (Tous les nœuds gérés) : tous les nœuds gérés dans les UO et les régions sélectionnées.
- Specify the resource group (Spécifier le groupe de ressources) : choisissez le nom d'un groupe de ressources dans la liste pour cibler les ressources qui lui sont associées.

> **Note**
>
> Actuellement, la sélection de groupes de ressources n'est prise en charge que pour les configurations à compte unique. Pour appliquer des correctifs aux ressources de plusieurs comptes, choisissez une autre option de ciblage.

- Specify a node tag (Spécifier une balise de nœud) : seuls les nœuds balisés avec la paire clé-valeur que vous spécifiez sont corrigés dans tous les comptes et régions que vous avez ciblés.
- Manual (Manuel) : choisissez manuellement les nœuds gérés parmi tous les comptes et régions spécifiés dans une liste.

> **Note**
>
> Actuellement, cette option ne prend en charge que les instances Amazon EC2.

13. Dans la section Rate control (Contrôle de taux), procédez comme suit :

    - Pour Concurrency (Simultanéité), saisissez un nombre ou un pourcentage de nœuds sur lesquels exécuter la politique de correctifs en même temps.
    - Dans Error threshold (Seuil d'erreur), saisissez le nombre ou le pourcentage de nœuds susceptibles de rencontrer une erreur avant que la politique de correctifs n'échoue.

14. (Facultatif) Sélectionnez Ajouter les politiques IAM requises aux profils d'instance existants affectés à vos instances.

    Cette sélection applique les politiques IAM créées par cette configuration Quick Setup aux nœuds auxquels un profil d'instance (instances EC2) ou une fonction du service (nœuds activés par un système hybride) sont déjà affectés. Nous vous recommandons de sélectionner cette option lorsque vos nœuds gérés sont déjà associés à un profil d'instance ou à une fonction du service, mais que ce dernier ou cette dernière ne contient pas toutes les autorisations requises pour travailler avec Systems Manager.

    Votre sélection s'applique aux nœuds gérés créés ultérieurement dans les comptes et les régions auxquels cette configuration de politique de correctifs s'applique.

> **Important**
>
> Si vous ne cochez pas cette case, mais que vous souhaitez que Quick Setup applique des correctifs à vos nœuds gérés à l'aide de cette stratégie de correctif, vous devez procéder comme suit :
> Ajoutez des autorisations à votre profil d'instance IAM ou à votre Fonction du service IAM pour accéder au compartiment S3 créé pour votre politique de correctifs
> Balisez votre profil d'instance IAM ou votre fonction du service IAM avec une paire clé-valeur spécifique.
> Pour plus d'informations, veuillez consulter Cas 1 : utiliser votre propre profil d'instance ou fonction du service avec vos nœuds gérés au lieu d'un profil fourni par Quick Setup.

15. Choisissez Créer.

    Pour vérifier le statut d'application des correctifs une fois la politique de correctifs créée, vous pouvez accéder à la configuration depuis la page Quick Setup.

DevOpsConfiguration du gourou

Vous pouvez configurer rapidement les options de DevOps Guru en utilisantQuick Setup. Amazon DevOps Guru est un service basé sur le machine learning (ML) qui permet d'améliorer facilement les performances opérationnelles et la disponibilité d'une application. DevOpsGuru détecte les comportements différents des modèles de fonctionnement normaux afin que vous puissiez identifier les problèmes opérationnels bien avant qu'ils n'affectent vos clients. DevOpsGuru ingère automatiquement les données opérationnelles de vos AWS applications et fournit un tableau de bord unique pour visualiser les problèmes liés à vos données opérationnelles. Vous pouvez commencer à utiliser DevOps Guru pour améliorer la disponibilité et la fiabilité des applications sans aucune expertise en matière de configuration manuelle ni d'apprentissage automatique.

La configuration de DevOps Guru avec Quick Setup est disponible dans les versions suivantes Régions AWS :

- USA Est (Virginie du Nord)
- USA Est (Ohio)
- USA Ouest (Oregon)
- Europe (Francfort)

- Europe (Irlande)
- Europe (Stockholm)
- Asie-Pacifique (Singapour)
- Asie-Pacifique (Sydney)
- Asia Pacific (Tokyo)

Pour plus d'informations sur les tarifs, consultez les tarifs Amazon DevOps Guru.

Prérequis

La région d'origine pour Quick Setup doit déjà être spécifiée avant de terminer les tâches suivantes. Pour plus d'informations, veuillez consulter Configuration de la Région AWS d'accueil.

Pour configurer DevOps Guru, effectuez les tâches suivantes dans la AWS Systems Manager Quick Setup console.

Pour configurer DevOps Guru avec Quick Setup

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Quick Setup.
3. Sur la carte DevOps Guru, choisissez Create.

   > **Tip**
   >
   > Si vous avez déjà une ou plusieurs configurations dans votre compte, choisissez d'abord l'onglet Bibliothèque ou le bouton Créer dans la section Configurations pour afficher les cartes.

4. Dans la section Configuration options (Options de configuration), sélectionnez les types de ressources AWS que vous voulez analyser, ainsi que vos préférences de notification.

   Si vous ne sélectionnez pas l'option Analyser toutes les AWS ressources de tous les comptes de mon organisation, vous pouvez choisir les AWS ressources à analyser ultérieurement dans la console DevOps Guru. DevOpsGuru analyse différents types de AWS ressources (tels que les buckets Amazon Simple Storage Service (Amazon S3) et les instances Amazon Elastic Compute Cloud (Amazon EC2)), qui sont classés en deux groupes de tarification. Vous payez les heures de ressources AWS analysées, pour chaque ressource active. Une ressource n'est active que si

elle produit des métriques, des événements ou des entrées de journal dans un délai d'une heure. Le tarif qui vous est facturé pour un type de AWS ressource spécifique dépend du groupe de prix.

Si vous sélectionnez l'option Activer les notifications SNS, une rubrique Amazon Simple Notification Service (Amazon SNS) est créée Compte AWS dans chacune des unités organisationnelles (UO) que vous ciblez avec votre configuration. DevOpsGuru utilise le sujet pour vous informer des événements importants du DevOps Guru, tels que la création d'un nouvel aperçu. Si vous n'activez pas cette option, vous pourrez ajouter un sujet ultérieurement dans la console DevOps Guru.

Si vous sélectionnez l' AWS Systems Manager OpsItemsoption Activer, des éléments de travail opérationnels (OpsItems) seront créés pour les EventBridge événements Amazon et les CloudWatch alarmes Amazon associés.

5. Dans la section Planification, sélectionnez la fréquence à laquelle vous voulez que Quick Setup corrige les modifications apportées aux ressources qui diffèrent de votre configuration. L'option Défaut (Par défaut) s'exécute une seule fois. Si vous ne voulez pas que Quick Setup corrige les modifications apportées aux ressources qui diffèrent de votre configuration, sélectionnez Disabled (Désactivé) sous Custom (Personnalisé).
6. Dans la section Cibles, choisissez d'autoriser DevOps Guru à analyser les ressources de certaines de vos unités organisationnelles (UO) ou du compte auquel vous êtes actuellement connecté.

   Si vous sélectionnez Custom (Personnalisé), passez à l'étape 8.

   Si vous sélectionnez Compte actuel, passez à l'étape 9.
7. Dans les sections UO cibles et Régions cibles, cochez les cases des UO et des régions dans lesquelles vous souhaitez utiliser DevOps Guru.
8. Choisissez les régions dans lesquelles vous souhaitez utiliser DevOps Guru sur le compte courant.
9. Sélectionnez Create (Créer).

Déploiement du package Distributor

Distributorest une capacité de AWS Systems Manager. Un package Distributor est un ensemble de logiciels ou de ressources installables qui peuvent être déployés en tant qu'une entité unique. AvecQuick Setup, vous pouvez déployer un Distributor package dans une Compte AWS et une

Région AWS ou plusieurs organisations dans AWS Organizations. Actuellement, seuls l'agent EC2Launch v2, le package d'utilitaires Amazon Elastic File System (Amazon EFS) et l' CloudWatch agent Amazon peuvent être déployés avec. Quick Setup Pour plus d'informations sur Distributor, consultez AWS Systems Manager Distributor.

Prérequis

La région d'origine pour Quick Setup doit déjà être spécifiée avant de terminer les tâches suivantes. Pour plus d'informations, veuillez consulter Configuration de la Région AWS d'accueil.

Pour déployer Distributor des packages, effectuez les tâches suivantes dans la AWS Systems Manager Quick Setup console.

Pour déployer des packages Distributor avec Quick Setup

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Quick Setup.
3. Sur la carte Distributeur, sélectionnez Créer.

   > **Tip**
   >
   > Si vous avez déjà une ou plusieurs configurations dans votre compte, choisissez d'abord l'onglet Bibliothèque ou le bouton Créer dans la section Configurations pour afficher les cartes.

4. Dans la section Configuration options (Options de configuration), sélectionnez le package que vous voulez déployer.
5. Dans la section Targets (Cibles), sélectionnez si vous voulez déployer le package dans l'ensemble de l'organisation, certaines unités organisationnelles (UO) ou le compte auquel vous êtes connecté.

   Si vous sélectionnez Entire organization (Ensemble de l'organisation), passez à l'étape 8.

   Si vous sélectionnez Personnalisé, passez à l'étape 7.
6. Dans la section Target OU (UO cibles), cochez les cases des UO et des régions dans lesquelles vous voulez déployer le package.
7. Sélectionnez Create (Créer).

Planification des ressources de l'instance Amazon EC2

Grâce à cette fonctionnalité Quick Setup AWS Systems Manager, vous pouvez configurer le Resource Scheduler pour automatiser le démarrage et l'arrêt des instances Amazon Elastic Compute Cloud (Amazon EC2).

Cette configuration Quick Setup vous permet de réduire les coûts de fonctionnement en démarrant et en arrêtant les instances selon la planification que vous spécifiez. Cette fonctionnalité vous permet d'éviter les coûts inutiles liés à l'exécution d'instances lorsqu'elles ne sont pas nécessaires. Par exemple, il est possible que vous laissiez vos instances s'exécuter en permanence, même si elles ne sont utilisées que 10 heures par jour, 5 jours par semaine. Vous pouvez plutôt planifier l'arrêt de vos instances tous les jours après les heures ouvrables. Ainsi, vous obtiendrez 70 % d'économies pour ces instances, car la durée d'exécution passerait de 168 à 50 heures. Aucun frais d'utilisation Quick Setup. Toutefois, des coûts peuvent être encourus en fonction des ressources configurées et des limites d'utilisation, sans frais pour les services utilisés pour votre configuration.

Avec le planificateur de ressources, vous pouvez choisir d'arrêter et de démarrer automatiquement des instances sur plusieurs instances Régions AWS et Comptes AWS selon un calendrier que vous définissez. La configuration Quick Setup cible les instances Amazon EC2 à l'aide de la clé et de la valeur de balise que vous spécifiez. Seules les instances dont la balise correspond à la valeur que vous spécifiez dans votre configuration sont arrêtées ou démarrées par le Planificateur de ressources.

Une configuration individuelle permet de planifier jusqu'à 5 000 instances par région. Si votre situation nécessite la planification de plus de 5 000 instances dans une région donnée, vous devez créer plusieurs configurations. Balisez vos instances en conséquence afin que chaque configuration gère jusqu'à 5 000 instances. Lorsque vous créez plusieurs configurations Quick Setup de Planificateur de ressources, vous devez spécifier différentes valeurs de clé de balise. Par exemple, une configuration peut utiliser la clé de balise « Env » avec la valeur « Prod », tandis qu'une autre utilise « Env » et « Dev ».

Si vous supprimez votre configuration, les instances ne sont plus arrêtées ni démarrées selon la planification précédemment définie. Dans de rares cas, les instances peuvent ne pas s'arrêter ou démarrer correctement en raison d'échecs de fonctionnement de l'API.

Le Planificateur de ressources démarre les instances balisées uniquement si elles sont à l'état `stopped`. De même, les instances ne sont arrêtées que si elles sont à l'état `running`. Le Planificateur de ressources fonctionne selon un modèle piloté par les événements et ne démarre ou

n'arrête les instances qu'aux heures que vous spécifiez. Par exemple, vous créez une planification qui démarre les instances à 9 h. Le Planificateur de ressources démarre toutes les instances associées à la balise que vous spécifiez et qui sont à l'état `stopped` à 9 h. Si les instances sont arrêtées manuellement ultérieurement, le Planificateur de ressources ne les redémarrera pas pour maintenir l'état `running`. De même, si une instance est démarrée manuellement après avoir été arrêtée conformément à votre planification, le Planificateur de ressources n'arrêtera pas l'instance à nouveau.

Si vous créez une planification avec une heure de début ultérieure à l'heure de fin, le Planificateur de ressources suppose que vos instances s'exécutent pendant la nuit. Par exemple, vous créez une planification qui démarre les instances à 21 h 00 et les arrête à 7 h 00. Le Planificateur de ressources démarre toutes les instances associées à la balise que vous spécifiez et qui sont à l'état `stopped` à 21 h 00 et les arrête à 7 h 00 le lendemain. Pour les planifications de nuit, l'heure de début s'applique aux jours que vous sélectionnez pour votre planification. Toutefois, l'heure d'arrêt s'applique au jour suivant de votre planification.

Prérequis

La région d'origine pour Quick Setup doit déjà être spécifiée avant de terminer les tâches suivantes. Pour plus d'informations, veuillez consulter Configuration de la Région AWS d'accueil.

Pour configurer la planification pour les instances Amazon EC2, effectuez les tâches suivantes dans la AWS Systems Manager Quick Setup console.

Configurer la planification des instances avec Quick Setup

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Quick Setup.
3. Sur la carte Planificateur de ressources, sélectionnez Créer.

   > **Tip**
   >
   > Si vous avez déjà une ou plusieurs configurations dans votre compte, choisissez d'abord l'onglet Bibliothèque ou le bouton Créer dans la section Configurations pour afficher les cartes.

4. Dans la section Instance tag (Balise d'instance), spécifiez la clé et la valeur de balise appliquées aux instances que vous souhaitez associer à votre planification.

5. Dans la section Schedule options (Options de planification), spécifiez le fuseau horaire, les jours et les heures auxquels vous souhaitez démarrer et arrêter vos instances.
6. Dans la section Targets (Cibles), choisissez si la planification doit être configurée sur un groupe Custom (Personnalisé) d'unités organisationnelles (UO) ou sur le Current account (Compte actuel) auquel vous êtes connecté :

   - Custom (Personnalisé) : dans la section Target OUs (UO cibles), sélectionnez les UO dans lesquelles vous souhaitez configurer la planification. Ensuite, dans la section Target Regions (Régions cibles), sélectionnez les régions dans lesquelles vous souhaitez configurer la planification.
   - Compte actuel : sélectionnez Current Region (Région actuelle) ou Choose Regions (Choisir des régions). Si vous avez sélectionné Choose Regions (Choisir les régions), choisissez les Target Regions (Régions cibles) dans lesquelles vous souhaitez configurer la planification.
7. Vérifiez les informations de planification dans la section Summary (Résumé).
8. Sélectionnez Create (Créer).

Explorateur de ressources AWS configuration

Grâce à Quick Setup cette fonctionnalité AWS Systems Manager, vous pouvez configurer rapidement Explorateur de ressources AWS pour rechercher et découvrir des ressources au sein de votre AWS organisation Compte AWS ou dans l'ensemble de celle-ci. Vous pouvez rechercher vos ressources à l'aide de métadonnées telles que les noms, les balises et les identifiants. Explorateur de ressources AWS fournit des réponses rapides à vos requêtes de recherche en utilisant des index. Resource Explorer crée et gère des index à l'aide de diverses sources de données pour recueillir des informations sur les ressources de votre Compte AWS.

Quick Setupfor Resource Explorer automatise le processus de configuration de l'index. Pour plus d'informations Explorateur de ressources AWS, voir Qu'est-ce que c'est Explorateur de ressources AWS ? dans le guide de Explorateur de ressources AWS l'utilisateur.

Pendant Quick Setup ce temps, Resource Explorer effectue les opérations suivantes :

- Crée un index Région AWS dans chaque élément de votre Compte AWS.
- Met à jour l'index dans la région que vous spécifiez comme indice agrégateur pour le compte.
- Crée une vue par défaut dans l'index de l'agrégateur Region. Cette vue ne comporte aucun filtre et renvoie donc toutes les ressources présentes dans l'index.

Autorisations minimales

Pour effectuer les étapes de la procédure suivante, vous devez disposer des autorisations suivantes :

- Action : `resource-explorer-2:*` — Ressource : aucune ressource spécifique (`*`)
- Action : `iam:CreateServiceLinkedRole` — Ressource : aucune ressource spécifique (`*`)

Pour configurer l'explorateur de ressources

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Quick Setup.
3. Choisissez une région d'origine, puis choisissez Commencer.
4. Sur la carte Resource Explorer, choisissez Create.
5. Dans la section Région de l'index agrégateur, choisissez la région dans laquelle vous souhaitez contenir l'index agrégateur. Vous devez sélectionner la région correspondant à l'emplacement géographique de vos utilisateurs.
6. (Facultatif) Cochez la case Remplacer les index d'agrégation existants dans des régions autres que celle sélectionnée ci-dessus.
7. Dans la section Cibles, choisissez l'organisation cible ou des unités organisationnelles (UO) spécifiques contenant les ressources que vous souhaitez découvrir.
8. Dans la section Régions, choisissez les régions à inclure dans la configuration.
9. Consultez le résumé de configuration, puis choisissez Create.

Sur la page Resource Explorer, vous pouvez surveiller l'état de la configuration.

Résolution des problèmes liés aux résultats Quick Setup

Échec du déploiement

Un déploiement échoue si l'ensemble de piles CloudFormation a échoué pendant la création. Suivez les étapes suivantes pour enquêter sur un échec de déploiement.

1. Accédez à la console AWS CloudFormation.

2. Sélectionnez la pile créée par votre configuration Quick Setup. Le Stack name (Nom de la pile) comprend `QuickSetup` suivi du type de configuration que vous avez choisi, par exemple, `SSMHostMgmt`.

   > **Note**
   >
   > CloudFormation supprime parfois les déploiements de pile qui ont échoué. Si la pile n'est pas disponible dans le tableau Stacks (Piles), sélectionnez Deleted (Supprimé) dans la liste des filtres.

3. Affichez les éléments Status (Statut) et Status reason (Raison du statut). Pour plus d'informations sur les statuts de pile, consultez Codes de statut de la pile dans le Guide de l'utilisateur AWS CloudFormation.
4. Pour comprendre l'étape exacte qui a échoué, consultez l'onglet Events (Événements) et passez en revue chaque Status (Statut) d'événement.
5. Consultez la section Dépannage dans le Guide de l'utilisateur AWS CloudFormation.
6. Si vous ne parvenez pas à résoudre l'échec du déploiement à l'aide des étapes de dépannage de CloudFormation, supprimez la configuration et redéfinissez-la.

Échec d'association

Le tableau Configuration details (Détails de la configuration) dans la page Configuration details (Détails de la configuration) de vos paramètres affichera Failed (Échec) comme Configuration status (Statut de configuration) si l'une des associations a échoué pendant la configuration. Procédez comme suit pour résoudre les problèmes d'une association ayant échoué.

1. Dans Configuration details (Détails de la configuration), sélectionnez la configuration qui a échoué, puis sélectionnez View details (Afficher les détails).
2. Copiez le Association name (Nom de l'association).
3. Accédez à State Manager et collez le nom de l'association dans le champ de recherche.
4. Sélectionnez l'association et sélectionnez l'onglet Execution history (Historique d'exécution).
5. Sous Execution ID (ID d'exécution), sélectionnez l'exécution d'association qui a échoué.
6. La page Association execution targets (Cibles d'exécution d'association) répertorie tous les nœuds sur lesquels l'association s'est exécutée. Sélectionnez le bouton Output (Sortie) pour une exécution dont l'exécution a échoué.

7. Sur la page Output (Sortie), sélectionnez 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 essayer de recréer l'association. Pour recréer l'association, supprimez d'abord l'association défaillante dans State Manager. Après avoir supprimé l'association, modifiez la configuration et sélectionnez l'option que vous avez supprimée, puis Update (Mettre à jour).

> **Note**
>
> Pour enquêter sur les associations au statut Failed (Échec) pour une configuration Organization (Organisation), vous devez vous connecter au compte avec l'association ayant échoué et utiliser la procédure d'association ayant échoué suivante, décrite précédemment. L'ID d'association n'est pas un hyperlien vers le compte cible lors de l'affichage des résultats à partir du compte de gestion.

Statut de l'écart

Lorsque vous consultez la page détaillée d'une configuration, vous pouvez afficher le statut de l'écart de chaque déploiement. Une dérive de configuration se produit chaque fois qu'une modification apportée par un utilisateur à un service ou une fonction entre en conflit avec les sélections effectuées via Quick Setup. Si une association a changé après la configuration initiale, le tableau affiche une icône d'avertissement indiquant le nombre d'éléments présentant des écarts. Vous pouvez déterminer la cause de l'écart en survolant l'icône.

Lorsqu'une association est supprimée dans State Manager, les déploiements associés affichent un avertissement d'écart. Pour résoudre ce problème, modifiez la configuration et sélectionnez l'option qui a été supprimée lorsque l'association a été supprimée. Sélectionnez Update (Mettre à jour) et attendez la fin du déploiement.

Gestion des opérations

Gestion des opérations est un ensemble de fonctionnalités qui vous aident à gérer vos ressources AWS .

Rubriques

- AWS Systems Manager Incident Manager
- AWS Systems Manager Explorer
- AWS Systems Manager OpsCenter
- CloudWatchTableaux de bord Amazon hébergés par Systems Manager

AWS Systems Manager Incident Manager

Utilisez Incident Manager, une fonctionnalité d'AWS Systems Manager, pour gérer les incidents qui se produisent dans vos applications hébergées par AWS. Incident Manager associe l'engagement des utilisateurs, l'escalade, les runbooks, les plans de réponse, les canaux de chat et l'analyse post-incident pour aider votre équipe à trier les incidents plus rapidement et à rétablir le fonctionnement normal de vos applications. Pour en savoir plus sur Incident Manager, veuillez consulter le Guide de l'utilisateur Incident Manager.

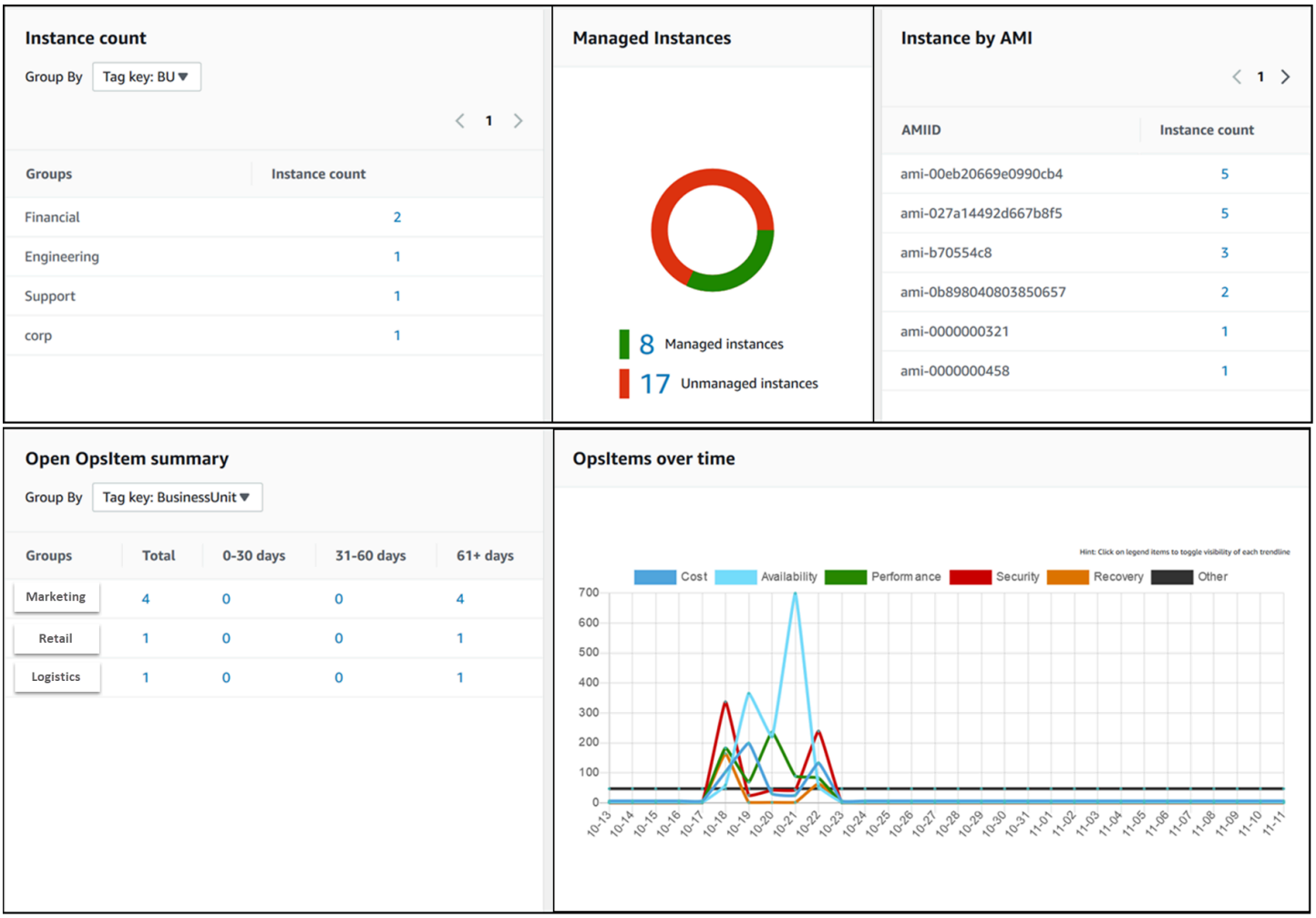
AWS Systems Manager Explorer

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 AWS et pour toutes les Régions AWS. Dans Explorer, OpsData inclut des métadonnées concernant les nœuds gérés dans votre environnement hybride et multicloud. OpsData inclut également des informations fournies par d'autres fonctionnalités de Systems Manager, notamment des informations sur la conformité aux correctifs Patch Manager et sur la conformité des associations State Manager. Pour simplifier davantage la façon dont vous accédez à OpsData, Explorer affiche des informations provenant de services AWS de support tels qu'AWS Config, AWS Trusted Advisor, AWS Compute Optimizer et AWS Support (dossiers de support).

Pour accroître la sensibilisation opérationnelle, Explorer affiche également des éléments de travail opérationnels (OpsItems). Explorer fournit un contexte sur la répartition des OpsItems entre vos unités commerciales ou vos applications, leur évolution sur la durée 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 la fonction OpsCenter de Systems Manager pour exécuter des runbooks Automation et résoudre rapidement ces problèmes. Pour vos premiers pas dans Explorer, ouvrez Systems Manager console. Dans le panneau de navigation, sélectionnez Explorer.

L'image suivante montre certaines des zones de rapport individuelles, appelées widgets, qui sont disponibles dans Explorer.



Quelles sont les fonctions d'Explorer ?

Explorer inclut les fonctionnalités 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, Patch Manager, State Manager, et de Services AWS de support tels que AWS Trusted Advisor, AWS Compute Optimizer et AWS Support. Ces widgets fournissent un contexte important pour vous aider à comprendre l'état et les risques opérationnels de vos ressources AWS. Voici quelques exemples de widgets d'information : Instance count (Nombre d'instances), Instance by AMI (Instance par AMI), Total noncompliant nodes (Nombre total de nœuds non conformes) (application de correctifs), Noncompliant associations (Associations non conformes) et Support Center cases (Dossiers du centre de support).
  - Widgets OpsItem : un OpsItem Systems Manager est un élément de travail opérationnel lié à une ou plusieurs ressources AWS. Les OpsItems sont une fonctionnalité de Systems ManagerOpsCenter. OpsItems peut exiger des ingénieurs DevOps qu'ils examinent et résolvent éventuellement un problème. Les OpsItems peuvent, par exemple, être l'utilisation élevée par une instance EC2 du processeur, les volumes détachés Amazon Elastic Block Store (Amazon EBS), un échec du déploiement AWS CodeDeploy ou un échec d'exécution de Systems Manager Automation. Parmi les exemples de widgets OpsItem, citons Open OpsItem summary (Résumé des OpsItems ouverts), OpsItem by status (OpsItem par état) et OpsItems over time (OpsItem dans le temps).
- Filters (Filtres) : chaque widget offre la possibilité de filtrer les informations en fonction du Compte AWS, de la Région AWS et de la balise. Les filtres vous aident à affiner rapidement les informations affichées dans Explorer.
- Direct links to service screens (Liens directs vers les écrans de service) : pour vous aider à analyser les problèmes liés aux ressources AWS, les widgets Explorer 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.
- Groups (Groupes) : pour vous aider à comprendre les types de problèmes opérationnels dans 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 les OpsItems.
- Trois modes d'affichage du Compte AWS et de la Région AWS : Explorer propose les modes d'affichage suivants pour les OpsData et les OpsItems dans les Comptes AWS et Régions AWS :
  - 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.

- Un seul compte/plusieursrégions : ce mode nécessite la création d'une ou plusieurs synchronisations de données de ressources sur la page Paramètres d'Explorer. Une synchronisation de données de ressource agrège les données opérationnelles provenant d'une ou de 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.
- Multiple-account/multiple-Region (Plusieurs comptes/plusieurs régions) : ce mode nécessite que votre organisation ou votre entreprise utilise AWS Organizations avec l'option All features (Toutes les fonctions) activée. Après avoir configuré AWS Organizations dans votre environnement informatique, vous pouvez agréger toutes les données de compte dans un compte de gestion. 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 All features (Toutes les fonctions) d'Organizations, consultez Activation du mode All Features dans votre organisation.

- Génération de rapports : vous pouvez exporter les rapports Explorer sous forme de fichiers séparés par des virgules (.csv) vers un compartiment Amazon Simple Storage Service (Amazon S3). Vous recevez une alerte d'Amazon Simple Notification Service (Amazon SNS) lorsqu'une exportation est terminée.

Quel est le lien entre Explorer et OpsCenter ?

Systems Manager OpsCenter offre un emplacement centralisé dans lequel les ingénieurs d'exploitation et les professionnels de l'informatique voient, étudient et résolvent les OpsItems associées aux ressources AWS. Explorer est un hub de rapports où les gestionnaires DevOps consultent des résumés agrégés de leurs données opérationnelles, notamment les OpsItems dans les comptes et Régions AWS. Explorer aide les utilisateurs à détecter des tendances et des modèles et, le cas échéant, 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 de plus amples informations, veuillez consulter Mise en route avec Systems Manager Explorer et OpsCenter.

Que sont les données opérationnelles, OpsData ?

OpsData, ce sont toutes les données opérationnelles affichées dans le tableau de bord Systems Manager Explorer. Explorer extrait des OpsData à partir des sources suivantes :

- Amazon Elastic Compute Cloud (Amazon EC2)

  Les données affichées dans Explorer sont notamment les suivantes : nombre total de nœuds, nombre total de nœuds gérés et non gérés, et nombre de nœuds qui utilisent une 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 sévérité, 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 Patch Manager

  Les données affichées dans Explorer incluent le nombre de nœuds non conformes et de nœuds critiques non conformes.

- AWS Trusted Advisor

  Les données affichées dans Explorer incluent : le statut 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.

- Cas AWS Support Center

  Les données affichées dans Explorer comprennent : l'ID, la sévérité, le statut, l'heure de création, l'objet, le service et la catégorie du cas.

- AWS Config

  Les données affichées dans Explorer comprennent : un résumé global des règles AWS Config conformes et non conformes, le nombre de ressources conformes et non conformes, et des détails

spécifiques sur chacune d'entre elles (lorsque vous explorez une règle ou une ressource non conforme).

- AWS Security Hub

  Les données affichées dans Explorer comprennent : un résumé global des résultats de Security Hub, le nombre de chaque type de résultats groupés par sévérité, et des détails spécifiques sur le résultat.

> **Note**
>
> Pour voir des cas AWS Trusted Advisor et AWS Support Center dans Explorer, vous devez disposer d'un compte Entreprise ou Business paramétré avec AWS Support.

Vous pouvez afficher et gérer les sources d'OpsData à partir de la page Paramètres d'Explorer. Pour de plus amples informations sur la configuration des services qui remplissent des widgets Explorer avec des OpsData, consultez Configuration des services connexes.

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 plus d'informations, consultez AWS Systems Manager Pricing (Tarification CTlong).

Rubriques

- Mise en route avec Systems Manager Explorer et OpsCenter
- Utilisation de Systems Manager Explorer
- Exportation OpsData depuis Systems Manager Explorer
- Résolution des problèmes liés à Systems Manager Explorer

Mise en route avec Systems Manager Explorer et OpsCenter

AWS 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.

> **Note**
>
> La configuration intégrée n'est disponible que dans la console Systems Manager. Vous ne pouvez pas paramétrer Explorer ou OpsCenter par programmation.

L'installation intégrée effectue les tâches suivantes :

- Configure les rôles et les autorisations : l'installation intégrée crée un rôle AWS Identity and Access Management (IAM) qui permet à Amazon EventBridge de créer automatiquement des OpsItems selon des règles par défaut. Après la configuration, vous devez configurer les autorisations d'utilisateur, de groupe ou de rôle pour OpsCenter, comme décrit dans cette section.
- Allows default rules for OpsItem creation (Autorise les règles par défaut pour la création d'OpsItems) : l'installation intégrée crée les 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.
- Allows OpsData sources (Autorise les sources d'OpsData) : l'installation intégrée autorise les sources de données qui renseignent les widgets Explorer.
- Allows you to specify reporting tag keys (Vous permet de spécifier des clés de balise de génération de rapports) : l'installation intégrée vous permet de spécifier jusqu'à cinq clés de balise de génération de rapports à affecter automatiquement à de nouveaux OpsItems qui répondent à des critères spécifiques.

Après avoir terminé l'installation intégrée, nous vous recommandons de Configurer Explorer de sorte à afficher les données de plusieurs régions et comptes. Explorer et OpsCenter synchronisent automatiquement OpsData et OpsItems pour le Compte AWS et la Région AWS que vous avez

utilisés lors de l'installation intégrée. Vous pouvez agréger les OpsData et OpsItems 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

AWS Systems Manager Explorer et AWS Systems Manager OpsCenter collectent des informations auprès, ou interagissent avec d'autres Services AWS et fonctionnalités de Systems Manager. 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 vous permettent de résoudre les problèmes avec les ressource s AWS à partir à l'aide de runbooks Systems Manager Automation. Pour utiliser cette fonctionnalité de correction, vous devez disposer d'une autorisation pour exécuter des documents Systems Manager Automation. Pour de plus amples informations, veuillez consulter Configuration d'Automation.
Installation et configuration de Systems Manager Patch Manager	Explorer comprend un widget qui fournit des informations sur la conformité des correctif s. Pour afficher ces données dans Explorer, vous devez configurer l'application de correctif s. Pour de plus amples informations, veuillez consulter AWS Systems Manager Patch Manager.

Tâche	Informations
Installation et configuration de Systems Manager State Manager	Explorer comprend un widget qui fournit des informations sur la conformité des associati ons Systems Manager State Manager. Pour afficher ces données dans Explorer, vous devez configurer State Manager. Pour de plus amples informations, veuillez consulter AWS Systems Manager State Manager.
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 consulter ces données dans Explorer, activez l'enregis treur 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.

Tâche	Informations
Activer AWS Trusted Advisor	Explorer utilise les données fournies par Trusted Advisor pour afficher le statut 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, veuillez consulter AWS Support.
Activer AWS Compute Optimizer	Explorer utilise les données fournies par Compute Optimizer pour afficher les détails du nombre d'instances EC2 Under provisioned (Sous provisionnées) et Over provisioned (Surprovisionnées), les résultats d'optimisation, les détails de tarification à la demande et des recommandations pour le type d'instance et le prix. Pour consulter ces données dans Explorer, activez Compute Optimizer. Pour plus d'informations, consultez Démarrez avec AWS Compute Optimizer.
Activer AWS Security Hub	Explorer utilise les données fournies par Security Hub pour remplir les widgets avec des informations relatives à vos résultats de sécurité. Pour consulter ces données dans Explorer, activez l'intégration de Security Hub. Pour plus d'informations, consultez Qu'est-ce qu'AWS Security Hub ?.

Configuration des rôles et des autorisations pour Systems Manager Explorer

L'installation intégrée crée et configure automatiquement les rôles AWS Identity and Access Management (IAM) pour AWS Systems Manager Explorer et AWS Systems Manager 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

L'installation intégrée crée et configure les rôles suivants pour travailler avec Explorer et OpsCenter.

- `AWSServiceRoleForAmazonSSM` : fournit l'accès aux ressources gérées par AWS ou utilisées par Systems Manager.
- `OpsItem-CWE-Role` : autorise CloudWatch Events et EventBridge à créer des OpsItems en réponse à des événements courants.
- `AWSServiceRoleForAmazonSSM_AccountDiscovery` : autorise Systems Manager à appeler d'autres Services AWS pour découvrir des informations Compte AWS lors de la synchronisation des données. Pour plus d'informations sur ce rôle, consultez À propos du rôle `AWSServiceRoleForAmazonSSM_AccountDiscovery`.
- `AmazonSSMExplorerExport` : autorise Explorer à exporter des OpsData vers un fichier avec des valeurs séparées par des virgules (CSV).

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 politique 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 Compte AWS informations pour OpsCenter et Explorer.

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 afin que les utilisateurs puissent effectuer des actions dans OpsCenter.

Avant de commencer

Vous pouvez configurer votre OpsCenter pour créer et gérer les OpsItems sur plusieurs comptes ou un seul compte. Si vous configurez OpsCenter pour créer et gérer les OpsItems sur plusieurs comptes, le compte de gestion AWS Organizations peut créer, afficher ou modifier des OpsItems manuellement sur d'autres comptes. Si nécessaire, vous pouvez également sélectionner le compte d'administrateur délégué de Systems Manager pour créer et gérer les OpsItems sur les comptes membre. Toutefois, si vous configurez OpsCenter pour un seul compte, vous pouvez uniquement consulter ou modifier les OpsItems du compte sur lequel les OpsItems ont été créés. 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 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 opérations d'API suivantes. Vous pouvez utiliser toutes les fonctions d'Explorer et d'OpsCenter si votre utilisateur, groupe ou rôle 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

Si vous préférez, vous pouvez spécifier l'autorisation de lecture seule en ajoutant la politique en ligne suivante à votre compte, groupe ou rôle.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ssm:GetOpsItem",
        "ssm:GetOpsSummary",
        "ssm:DescribeOpsItems",
        "ssm:GetServiceSetting",
        "ssm:ListResourceDataSync"
      ],
      "Resource": "*"
    }
  ]
}
```

Pour de plus amples informations sur la création de politiques IAM, consultez Création de politiques IAM dans le guide de l'utilisateur IAM. Pour de plus amples informations sur la façon d'attribuer cette politique à un groupe IAM, consultez Attacher une politique à un groupe IAM.

Créez une autorisation à l'aide des éléments suivants et ajoutez-la à vos utilisateurs, groupes ou rôles :

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

En fonction de l'application d'identité que vous utilisez dans votre organisation, vous pouvez sélectionner n'importe laquelle des options suivantes pour configurer l'accès des utilisateurs.

Pour activer l'accès, ajoutez des autorisations à vos utilisateurs, groupes ou rôles :

- Utilisateurs et groupes dans AWS IAM Identity Center :

  Créez un jeu d'autorisations. Suivez les instructions de la rubrique Création d'un jeu d'autorisations du Guide de l'utilisateur AWS IAM Identity Center.
- Utilisateurs gérés dans IAM par un fournisseur d'identité :

  Créez un rôle pour la fédération d'identité. Pour plus d'informations, voir la rubrique Création d'un rôle pour un fournisseur d'identité tiers (fédération) du Guide de l'utilisateur IAM.
- Utilisateurs IAM :
  - Créez un rôle que votre utilisateur peut assumer. Suivez les instructions de la rubrique Création d'un rôle pour un utilisateur IAM du Guide de l'utilisateur IAM.

- (Non recommandé) Attachez une politique directement à un utilisateur ou ajoutez un utilisateur à un groupe d'utilisateurs. Suivez les instructions de la rubrique Ajout d'autorisations à un utilisateur (console) du Guide de l'utilisateur IAM.

Restriction de l'accès aux OpsItems à l'aide de balises

Vous pouvez également limiter l'accès de OpsItems à l'aide d'une politique IAM en ligne qui spécifie des balises. Voici un exemple qui spécifie une clé de balise Service et une valeur de balise Finance. Grâce à cette politique, l'utilisateur peut uniquement appeler l'opération d'API GetOpsItem pour afficher le OpsItems qui était précédemment balisé 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 opérations d'API pour l'affichage et la mise à jour d' OpsItems. Cette politique spécifie également deux ensembles de paires clé/valeur de balises : ministère 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 Créer manuellement OpsItems.

Activation des règles par défaut

L'installation intégrée configure automatiquement les règles par défaut suivantes dans Amazon EventBridge. Ces règles créent des OpsItems dans AWS Systems Manager OpsCenter. Si vous ne souhaitez pas qu'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 de plus amples informations, veuillez consulter Configurer des règles EventBridge pour créer des OpsItems. Vous pouvez également désactiver les règles par défaut à tout moment sur la page Settings (Paramètres).

> **Important**
>
> Actuellement, vous ne pouvez pas modifier les valeurs de Category (Catégorie) et Severity (Sévérité) pour les règles par défaut, mais vous pouvez modifier ces valeurs sur les OpsItems créés 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 sources OpsData

La configuration intégrée active les sources de données suivantes qui renseignent les widgets Explorer.

- AWS Support Center (Vous devez disposer d'un plan Business or Enterprise pour activer cette source.)
- AWS Compute Optimizer Center (Vous devez disposer d'un plan Business or Enterprise pour activer cette source.)
- Conformité des associations Systems Manager State Manager
- Conformité d'AWS Config
- Systems Manager OpsCenter
- Conformité des correctifs Systems Manager Patch Manager
- Amazon Elastic Compute Cloud (Amazon EC2)
- Systems Manager Inventory
- AWS Trusted Advisor Center (Vous devez disposer d'un plan Business or Enterprise pour activer cette source.)
- AWS Security Hub

Spécification des clés de balise

Lorsque vous configurez AWS Systems Manager Explorer, vous pouvez spécifier jusqu'à cinq clés d'identification 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

AWS Systems Manager utilise une expérience d'installation intégrée pour vous aider à démarrer avec AWS Systems Manager Explorer et AWS Systems Manager OpsCenter. Après avoir terminé l'installation intégrée, Explorer et OpsCenter synchronisent automatiquement les données. Plus précisément, ces fonctionnalités permettent de synchroniser OpsData et OpsItems pour le Compte AWS et la Région AWS que vous avez utilisés lorsque vous avez terminé l'installation intégrée. Si vous souhaitez agréger OpsData et OpsItems à partir d'autres comptes et régions, vous devez créer une synchronisation des données des ressources, comme décrit dans cette rubrique.

> **Note**
>
> Pour de plus amples informations sur la configuration intégrée, veuillez consulter Mise en route avec Systems Manager Explorer et OpsCenter.

À propos de la synchronisation des données des ressources pour Explorer

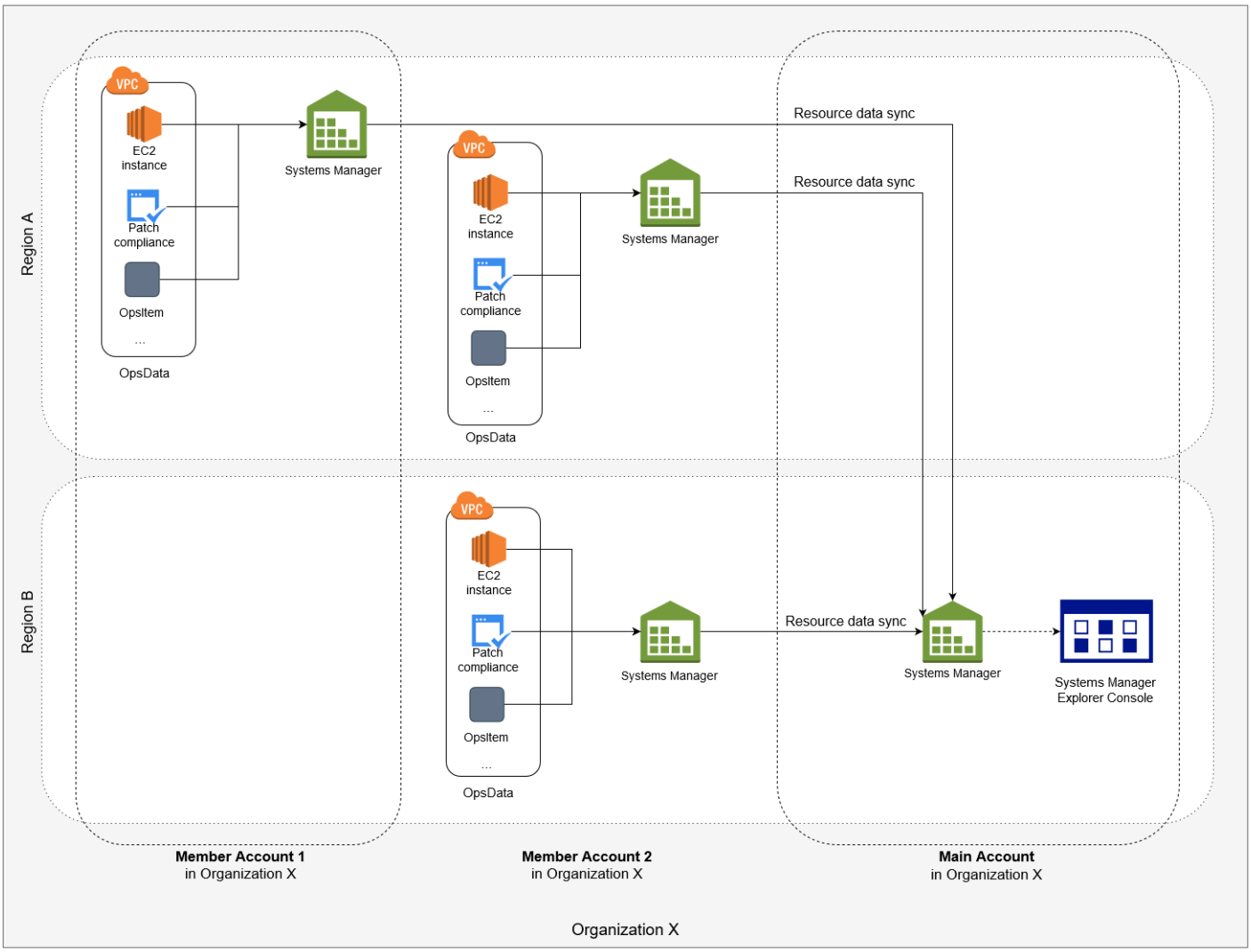
La synchronisation des données des ressources pour Explorer propose deux options d'agrégation :

- Single-account/Multiple-regions (Un seul compte/plusieurs régions) : vous pouvez configurer Explorer pour agréger des données OpsItems et opsData de plusieurs Régions AWS, mais l'ensemble de données est limité au Compte AWS actuel.
- Multiple-accounts/Multiple-regions (Plusieurs comptes/plusieurs régions) : vous pouvez configurer Explorer pour agréger les données de plusieurs comptes et Régions AWS. Cette option demande

la configuration d' AWS Organizations. Après l'installation et la configuration d'AWS Organizations, vous pouvez agréger les données dans l'Explorer par unité d'organisation ou pour une organisation entière. Systems Manager agrège les données dans le compte de gestion AWS Organizations avant de l'afficher dans Explorer. Pour plus d'informations, consultez Présentation d'AWS Organizations dans le Guide de l'utilisateur AWS Organizations.

> **Warning**
>
> Si vous configurez Explorer pour ajouter les données d'une organisation dans AWS Organizations, le système activera OpsData dans tous les comptes membres de l'organisation. L'activation des sources OpsData dans tous les comptes membres augmente le nombre d'appels des API OpsCenter tels que CreateOpsItem et GetOpsSummary. Les appels à ces actions d'API vous sont facturés.

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 des ressources regroupe les données des deux comptes et de plusieurs Régions AWS dans le compte principal AWS Organizations où elles sont ensuite affichées dans Explorer.



À propos de la synchronisation de données de ressources de plusieurs comptes et plusieurs régions

Cette section détaille les synchronisations de données de ressources de plusieurs comptes et plusieurs régions utilisant AWS Organizations. Les informations de cette section s'appliquent si vous sélectionnez spécifiquement l'une des options suivantes sur la page Création de synchronisation de données de ressources :

- Inclure tous les comptes à partir de ma configuration d'AWS Organizations
- Sélectionner des unités organisationnelles dans AWS Organizations

Si vous n'envisagez pas d'utiliser l'une de ces options, vous pouvez ignorer cette section.

Lorsque vous créez une synchronisation de données de ressources dans la console SSM, si vous sélectionnez l'une des options AWS Organizations, Systems Manager autorise alors

automatiquement toutes les sources OpsData dans les régions sélectionnées pour tous les Comptes AWS de votre organisation (ou dans les unités d'organisation sélectionnées). Par exemple, même si vous n'avez pas activé Explorer dans une région, si vous sélectionnez une option AWS Organizations pour la synchronisation de vos données de ressources, Systems Manager collecte automatiquement des OpsData à partir de cette région. Pour créer une synchronisation de données de ressources sans autoriser les sources OpsData, spécifiez EnableAllOpsDataSources sur false lors de la création de la synchronisation de données. Pour plus d'informations, consultez EnableAllOPSDataSources dans la Référence d'API Amazon EC2 Systems Manager.

Si vous ne sélectionnez pas l'une des options AWS Organizations pour une synchronisation des données de ressources, vous devez effectuer la configuration intégrée dans chaque compte et région où vous voulez qu'Explorer accède aux données. Dans le cas contraire, Explorer n'affiche pas les OpsData ni les OpsItems pour les comptes et régions dans lesquels vous n'avez pas effectué la configuration intégrée.

Si vous ajoutez un compte enfant à votre organisation, Explorerautorise automatiquement toutes les sources d'OpsData pour le compte. Si vous supprimez le compte enfant de votre organisation ultérieurement, Explorer continue de collecter des OpsData à partir du compte.

Si vous mettez à jour une synchronisation de données de ressources existante utilisant l'une des options AWS Organizations, le système vous invite à approuver la collecte de toutes les sources de OpsData pour tous les comptes et régions concernés par la modification.

Si vous ajoutez un nouveau service à votre Compte AWS, et si Explorer collecte des OpsData pour ce service, Systems Manager configure alors automatiquement Explorer pour collecter ces OpsData. Par exemple, si votre organisation n'a pas utilisé AWS Trusted Advisor lors de la création d'une synchronisation des données de ressources, mais qu'elle s'abonne à ce service, Explorer met automatiquement à jour vos données de ressources synchronisées pour collecter ces OpsData.

> **Important**
>
> Notez les informations suivantes. Elles sont importantes pour la synchronisation de données de ressources de plusieurs comptes et régions :
>
> - La suppression d'une synchronisation des données de ressources ne désactive pas une source OpsData dans Explorer.

- Pour afficher les OpsData et les OpsItems de plusieurs comptes, le mode AWS Organizations All features (Toutes les fonctions) doit être activé et vous devez être connecté au compte de gestion AWS Organizations.

Création d'une synchronisation des données de ressource

Avant de configurer la synchronisation des données de ressources pour Explorer, notez les détails suivants.

- Explorer prend en charge un maximum de cinq synchronisations de données de ressources.
- 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 AWS Organizations). Vous devez supprimer la première synchronisation des données de ressource et en créer une nouvelle. Pour de plus amples informations, consultez Suppression des données de ressource Systems Manager Explorer
- Les OpsData affichées dans Explorersont en lecture seule.

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, sélectionnez Explorer.
3. Sélectionnez Settings (Paramètres).
4. Dans la section Configure resource data sync (Configurer la synchronisation des données de ressource), sélectionnez 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) sélectionnez une option.

> **Note**
>
> Pour utiliser l'une des options AWS Organizations, vous devez être connecté au compte de gestion AWS Organizations ou vous devez être connecté à un compte administrateur délégué Explorer. Pour de plus amples informations sur le compte d'administrateur délégué, veuillez consulter Configuration d'un administrateur délégué.

7. Dans la section Regions to include (Régions à inclure) sélectionnez l'une des options suivantes.

   - Sélectionnez 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.
   - Sélectionnez All regions (Toutes les régions) pour synchroniser automatiquement les données de toutes les Régions AWS actuelles.
   - Sélectionnez individuellement les régions à inclure.

8. Sélectionnez 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 consulter 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 d'AWS Systems Manager Explorer provenant de plusieurs comptes et Régions AWS à l'aide de la synchronisation des données de ressources avec AWS Organizations, nous vous recommandons de configurer un administrateur délégué pour Explorer.

Un administrateur délégué peut utiliser les API de synchronisation des données de ressources Explorer suivantes à l'aide de la console, du kit SDK, de l'AWS Command Line Interface (AWS CLI) ou d'AWS Tools for Windows PowerShell :

- CreateResourceDataSync
- DeleteResourceDataSync
- ListResourceDataSync

- UpdateResourceDataSync

Un administrateur délégué peut créer cinq synchronisations des données de ressources au maximum pour une organisation entière ou un sous-ensemble d'unités organisationnelles. 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 le compte de gestion AWS Organizations.

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. Pour de plus amples informations sur AWS Organizations, consultez Qu'est-ce que AWS Organizations ? dans le Guide de l'utilisateur AWS Organizations.

Rubriques

- Configurer un administrateur délégué Explorer
- Désenregistrer un administrateur délégué Explorer

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 compte de gestion AWS Organizations.
2. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
3. Dans le panneau de navigation, sélectionnez Explorer.
4. Sélectionnez Settings (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, sélectionnez les boutons Create role (Créer un rôle) et Enable access (Activer l'accès) pour configurer ces options.
6. Dans Account ID (ID de compte), saisissez l'ID de Compte AWS. Ce compte doit être un compte membre dans AWS Organizations.
7. Sélectionnez Enregistrer l'administrateur délégué.

L'administrateur délégué a désormais accès aux options Include all accounts from my configuration AWS Organizations (Inclure tous les comptes de ma configuration) et Select organization units in AWS Organizations (Sélectionner des unités organisationnelles dans) sur la page Create resource data sync (Créer une synchronisation de données de ressources).

Désenregistrer un administrateur délégué Explorer

Pour annuler l'inscription d'un administrateur délégué Explorer, procédez comme suit. L'inscription d'un compte administrateur délégué ne peut être annulée que par le compte de gestion AWS Organizations. 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 compte de gestion AWS Organizations.
2. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
3. Dans le panneau de navigation, sélectionnez Explorer.
4. Sélectionnez Settings (Paramètres).
5. Dans la section Administrateur délégué pour Explorer, sélectionnez Désenregistrer. Le système affiche un avertissement.
6. Saisissez l'ID du compte et sélectionnez Remove (Supprimer).

Le compte n'a plus accès aux opérations 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 AWS Systems Manager Explorer en modifiant la disposition du widget et les données affichées dans le tableau de bord.

Table des matières

- Personnalisation de l'affichage et utilisation de filtres
- Suppression des données de ressource Systems Manager Explorer
- Recevoir des résultats de AWS Security Hub dans Explorer

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 AWS Systems Manager OpsCenter. AWS Systems Manager Explorer affiche ensuite des informations agrégées concernant les OpsItems.

Chaque règle comprend des valeurs Catégorie et Sévérité prédéfinies. Lorsque le système crée des OpsItems à partir d'un événement, il affecte automatiquement la Catégorie et la Sévéritéprédéfinies.

> **Important**
>
> Actuellement, vous ne pouvez pas modifier les valeurs de Category (Catégorie) et Severity (Sévérité) pour les règles par défaut, mais vous pouvez modifier ces valeurs sur les OpsItems créés 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, sélectionnez Explorer.
3. Sélectionnez Settings (Paramètres).
4. Dans la sectionRègles d'OpsItems, sélectionnez 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 Sévérité pour modifier ces informations pour une règle.
8. Sélectionnez Enregistrer.

Vos modifications prennent effet à la prochaine création par le système crée d'un OpsItem.

Modification de sources de données Systems Manager Explorer

AWS Systems Manager Explorer affiche les données provenant des sources suivantes. Vous pouvez modifier les paramètres d'Explorer pour ajouter ou supprimer des sources de données :

- Amazon Elastic Compute Cloud (Amazon EC2)
- OpsCenter AWS Systems Manager
- Conformité des correctifs AWS Systems Manager Patch Manager
- Conformité des associations AWS Systems Manager State Manager
- AWS Trusted Advisor
- AWS Compute Optimizer
- Cas AWS Support Center
- Conformité des règles et des ressources AWS Config
- Résultats AWS Security Hub

> **Note**
>
> - Pour consulter des cas AWS Support Center dans Explorer, vous devez disposer d'un compte Entreprise ou Business paramétré 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 avez installé et configuré les services qui remplissent les widgets Explorer avec des données. Pour de plus amples informations, veuillez consulter Configuration des services connexes.

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, sélectionnez Explorer.
3. Sélectionnez Settings (Paramètres).
4. Dans la section OpsData sources (Sources OpsData), sélectionnez Modifier.
5. Développez OpsData sources (Sources OpsData).
6. Ajouter ou supprimer une ou plusieurs sources.
7. Choisissez Enregistrer.

Personnalisation de l'affichage et utilisation de filtres

Vous pouvez personnaliser la disposition des widgets dans AWS Systems Manager Explorer grâce à la fonction de glisser-déposer. Vous pouvez également personnaliser les OpsData et les OpsItems affichés dans Explorer à l'aide de filtres, comme décrit dans cette rubrique.

Avant de commencer

Avant de personnaliser la disposition des widgets, vérifiez que les widgets que vous souhaitez afficher sont actuellement affichés dans Explorer. Pour afficher certains widgets dans Explorer (tels que le widget de conformité AWS Config), vous devez les activer sur la page Configure dashboard (Configurer le tableau de bord).

Activer l'affichage des widgets dans Explorer

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.

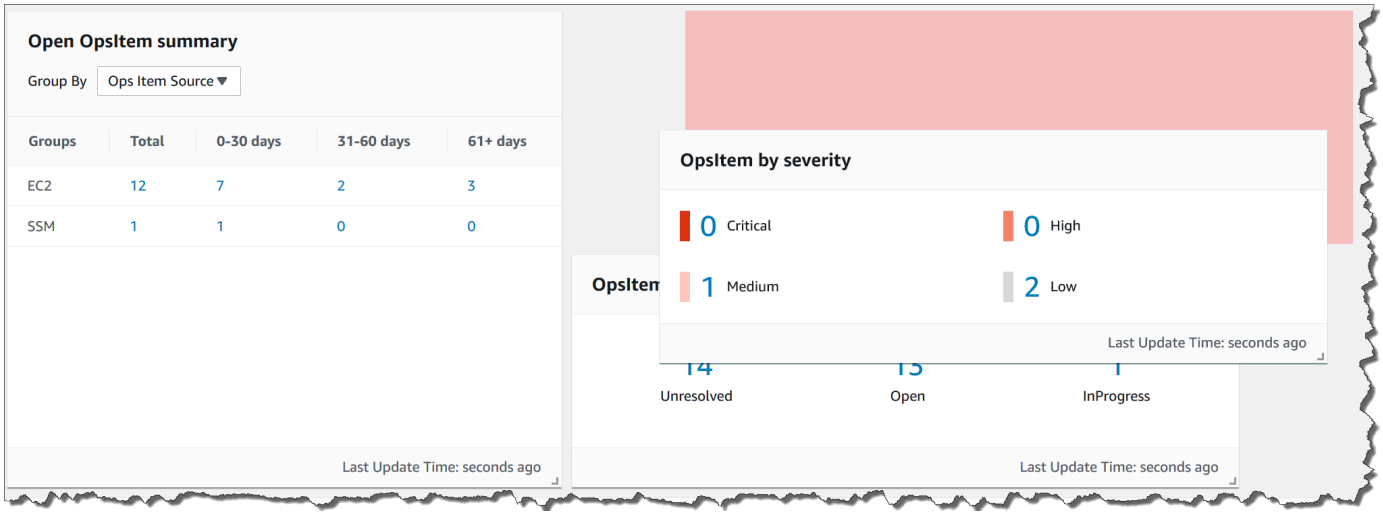
2. Dans le panneau de navigation, sélectionnez Explorer.
3. Choisissez Dashboard actions (Actions du tableau de bord), puis Configure dashboard (Configurer le tableau de bord).
4. Choisissez l'onglet Configure dashboard (Configurer le tableau de bord).
5. Choisissez Enable all (Tout activer) ou activez une source de données ou un widget individuel.
6. Choisissez Explorer pour consulter les modifications.

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, sélectionnez Explorer.
3. Sélectionnez 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.

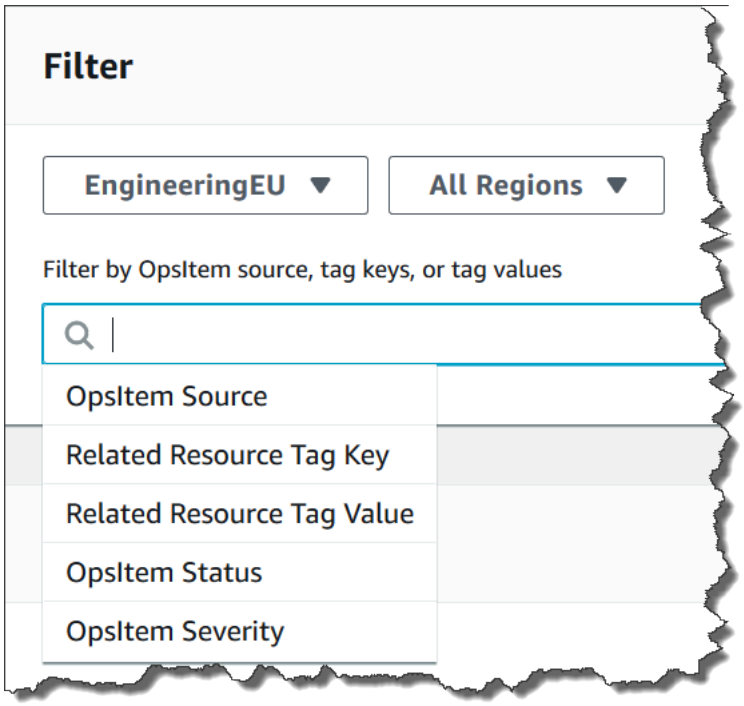
Si vous décidez que vous n'aimez pas la nouvelle disposition, sélectionnez Reset layout (Réinitialiser la disposition) pour déplacer tous les widgets à leur emplacement d'origine.

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 OpsItem et critères de balise-clé.

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, sélectionnez 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 Regions (Régions) pour choisir une Région AWS spécifique ou sélectionnez All regions (Toutes les régions).
5. Sélectionnez la barre de recherche, puis sélectionnez 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 des données de ressource Systems Manager Explorer

Dans AWS Systems Manager Explorer, vous pouvez agréger des OpsData et OpsItems 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 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, sélectionnez Explorer.
3. Sélectionnez Settings (Paramètres).
4. Dans la section Configure resource data sync (Configurer la synchronisation des données de ressource), sélectionnez la synchronisation des données de ressource que vous souhaitez supprimer.
5. Sélectionnez Delete.

Recevoir des résultats de AWS Security Hub dans Explorer

AWS Security Hubfournit une vue complète de votre état de sécurité dans AWS. Le service collecte des données de sécurité, appelées résultats, provenant de l'ensemble des Comptes AWS, des services et des produits tiers pris en charge. Les résultats de Security Hub peuvent vous aider à vérifier votre environnement par rapport aux normes et aux bonnes pratiques du secteur de la sécurité, à analyser vos tendances en matière de sécurité et à identifier les problèmes de sécurité les plus prioritaires.

Security Hub envoie les résultats à Amazon EventBridge, qui utilise une règle d'événement pour envoyer les résultats àExplorer. Après avoir activé l'intégration, comme décrit ici, vous pouvez afficher les résultats de Security Hub dans un widget Explorer et afficher les détails des résultats

dans OpsCenter OpsItems. Le widget fournit un résumé de toutes les résultats de Security Hub en fonction de la gravité. Les nouveaux résultats dans Security Hub sont généralement visibles dans Explorer quelques secondes après leur création.

Warning

Notez les informations importantes suivantes :

- Explorer est intégré à OpsCenter, une fonctionnalité de Systems Manager. Après avoir activé l'intégration de Explorer avec Security Hub, OpsCenter crée automatiquement OpsItems pour les résultats de Security Hub. En fonction de votre AWS environnement, l'activation de OpsItems l'intégration peut entraîner un grand nombre de

  Avant de continuer, lisez les informations sur l'intégration de OpsCenter avec Security Hub. Cette rubrique comprend des détails spécifiques sur la façon dont les modifications et les mises à jour des résultats et des OpsItems sont imputées à votre compte. Pour plus d'informations, consultez AWS Security Hub. Pour plus d'informations sur la tarification de OpsCenter, consultez Tarification de AWS Systems Manager.
- Si vous créez une synchronisation de données de ressources dans Explorer alors que vous êtes connecté au compte d'administrateur, l'intégration de Security Hub est automatiquement activée pour l'administrateur et tous les comptes membres dans la synchronisation. Une fois activée, OpsCenter crée automatiquement OpsItems pour les résultats de Security Hub, moyennant un coût. Pour plus d'informations sur la création d'une synchronisation des données de ressources, consultez Configuration de Systems Manager Explorer de sorte à afficher les données de plusieurs comptes et Régions.

Types de résultats reçus par Explorer

Explorer reçoit tous les résultats de Security Hub. Vous pouvez voir tous les résultats dans le widget Explorer, en fonction de la sévérité, lorsque vous activez les paramètres par défaut de Security Hub. Par défaut, Explorer crée OpsItems pour les résultats critiques et de gravité élevée. Vous pouvez configurer manuellement Explorer pour créer OpsItems pour les résultats de gravité moyenne et faible.

Bien qu'il Explorer ne crée pas OpsItems de résultats informatifs, vous pouvez consulter les données relatives aux opérations informatives (OpsData) dans le widget de synthèse des résultats de Security Hub. Explorercrée OpsData pour tous les résultats, quelle que soit leur gravité. Pour plus

d'informations sur les niveaux de gravité de Security Hub, consultez Gravité (français non garanti) dans la Référence d'API AWS Security Hub .

Activation de l'intégration

Cette section décrit comment activer et configurer Explorer pour commencer à recevoir les résultats de Security Hub.

Avant de commencer

Exécutez les tâches suivantes avant de configurer Explorer pour commencer à recevoir les résultats de Security Hub.

- Activez et configurez Security Hub. Pour de plus amples informations, veuillez consulter Configuration de Security Hub dans le Guide de l'utilisateur AWS Security Hub .
- Connectez-vous au compte AWS Organizations de gestion. Systems Manager doit accéder à AWS Organizations pour créer des OpsItems à partir des résultats de Security Hub. Une fois connecté au compte de gestion, vous êtes invité à sélectionner le bouton Activer l'accès sous l'onglet Configurer un tableau de bord Explorer, comme l'explique la procédure suivante. Si vous ne vous connectez pas au compte de AWS Organizations gestion, vous ne pouvez pas autoriser l'accès et ne Explorer pouvez pas créer OpsItems à partir des résultats du Security Hub.

Pour commencer à recevoir les résultats de Security Hub

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Explorer.
3. Sélectionnez Paramètres.
4. Sélectionnez l'onglet Configurer un tableau de bord.
5. Sélectionnez AWS Security Hub.
6. Sélectionnez le curseur Désactivé pour activer AWS Security Hub.

   Les résultats critiques et de gravité élevée sont affichés par défaut. Pour afficher les résultats de gravité moyenne et faible, sélectionnez le curseur Désactivé en regard de Moyenne,Faible.
7. Dans la section OpsItems créés par les résultats de Security Hub, sélectionnez Activer l'accès. Si vous ne voyez pas ce bouton, connectez-vous au compte de AWS Organizations gestion et revenez sur cette page pour sélectionner le bouton.

Comment afficher les résultats de Security Hub

La procédure suivante décrit la consultation des résultats Security Hub.

Pour consulter les résultats Security Hub

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Explorer.
3. Recherchez le widget Résumé des résultats AWS Security Hub . Cela affiche les résultats de Security Hub. Vous pouvez sélectionner un niveau de sévérité pour afficher une description détaillée de l'OpsItem correspondant.

Comment arrêter l'envoi des résultats

La procédure suivante décrit l'arrêt de la réception des résultats de Security Hub.

Pour arrêter l'envoi des résultats de Security Hub

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Explorer.
3. Sélectionnez Paramètres.
4. Sélectionnez l'onglet Configurer un tableau de bord.
5. Sélectionnez le curseur Activé pour désactiver AWS Security Hub.

> **Important**
>
> Si l'option permettant de désactiver les résultats de Security Hub est grisée dans la console, vous pouvez désactiver ce paramètre en exécutant la commande suivante dans le AWS CLI. Vous devez exécuter la commande lorsque vous êtes connecté au compte AWS Organizations de gestion ou au compte administrateur délégué de Systems Manager. Pour le `region` paramètre, spécifiez l' Région AWS endroit où vous souhaitez arrêter de recevoir les résultats du Security HubExplorer.

```
aws ssm update-service-setting --setting-id /ssm/opsdata/SecurityHub --setting-
value Disabled --region Région AWS
```

Voici un exemple :

```
aws ssm update-service-setting --setting-id /ssm/opsdata/SecurityHub --setting-
value Disabled --region us-east-1
```

Exportation OpsData depuis Systems Manager Explorer

Vous pouvez exporter 5 000 OpsData articles sous forme de fichier de valeurs séparées par des virgules (.csv) vers un bucket Amazon Simple Storage Service (Amazon S3) depuis Explorer. AWS Systems Manager Explorer utilise le runbook AWS-ExportOpsDataToS3d'automatisation pour exporter OpsData. Lorsque vous exportez OpsData, le système affiche la page du manuel d'automatisation dans laquelle vous pouvez spécifier des détails, tels que AssumeRole, le nom du compartiment Amazon S3, l'ARN de la rubrique SNS et les champs à exporter.

Pour exporter OpsData :

- Étape 1 : spécification d'une rubrique SNS
- Étape 2 : (Facultatif) configuration de l'exportation de données
- Étape 3 : Exporter OpsData

Étape 1 : spécification d'une rubrique SNS

Lorsque vous configurez l'exportation de données, vous devez spécifier une rubrique Amazon Simple Notification Service (Amazon SNS) qui existe dans le Région AWS même endroit où vous souhaitez exporter les données. Systems Manager envoie une notification à la rubrique Amazon SNS lorsqu'une exportation est terminée. Pour obtenir des informations sur la création d'une rubrique Amazon SNS, veuillez consulter la rubrique Création d'une rubrique Amazon SNS.

Étape 2 : (Facultatif) configuration de l'exportation de données

Vous pouvez configurer les paramètres d'exportation des données depuis la page Paramètres ou Exporter les données des opérations vers le compartiment S3.

Pour configurer l'exportation de données depuis Explorer

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Explorer.
3. Sélectionnez Settings (Paramètres).
4. Dans la section Configure data export (Configurer l'exportation de données) sélectionnez Edit (Modifier).
5. Pour charger le fichier d'exportation de données vers un compartiment Amazon S3 existant, sélectionnez Sélectionner un compartiment S3 existant, puis sélectionnez le compartiment dans la liste.

   Pour charger le fichier d'exportation de données vers un nouveau compartiment Amazon S3, sélectionnez Créer un nouveau compartiment S3, puis saisissez le nom que vous souhaitez utiliser pour le nouveau compartiment.

   > **Note**
   >
   > Vous ne pouvez modifier le nom du compartiment Amazon S3 et l'ARN de la rubrique Amazon SNS qu'à partir de la page sur laquelle vous avez configuré ces paramètres pour la première fois dans Explorer. Si vous configurez le compartiment Amazon S3 et l'ARN de la rubrique Amazon SNS depuis la page Paramètres, vous ne pouvez modifier ces paramètres que depuis la page Paramètres.

6. Pour Sélectionner un ARN de rubrique Amazon SNS, choisissez la rubrique à notifier une fois l'exportation terminée.
7. Choisissez Créer.

Étape 3 : Exporter OpsData

Lorsque vous exportez Explorer des données, Systems Manager crée un rôle AWS Identity and Access Management (IAM) nomméAmazonSSMExplorerExportRole. Le rôle utilise la politique IAM suivante.

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

```
        {
            "Sid": "OpsSummaryExportAutomationServiceRoleStatement1",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject"
            ],
            "Resource": [
               "arn:aws:s3:::{{ExportDestinationS3BucketName}}/*"
            ]
        },
        {
            "Sid": "OpsSummaryExportAutomationServiceRoleStatement2",
            "Effect": "Allow",
            "Action": [
                "s3:GetBucketAcl",
                "s3:GetBucketLocation"
            ],
            "Resource": [
               "arn:aws:s3:::{{ExportDestinationS3BucketName}}"
            ]
        },
        {
            "Sid": "OpsSummaryExportAutomationServiceRoleStatement3",
            "Effect": "Allow",
            "Action": [
                "sns:Publish"
            ],
            "Resource": [
                "{{SnsTopicArn}}"
            ]
        },
        {
            "Sid": "OpsSummaryExportAutomationServiceRoleStatement4",
            "Effect": "Allow",
            "Action": [
                "logs:DescribeLogGroups",
                "logs:DescribeLogStreams"
            ],
            "Resource": [
                "*"
            ]
        },
        {
            "Sid": "OpsSummaryExportAutomationServiceRoleStatement5",
```

```
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogGroup",
                "logs:PutLogEvents",
                "logs:CreateLogStream"
            ],
            "Resource": [
                "*"
            ]
        },
        {
            "Sid": "OpsSummaryExportAutomationServiceRoleStatement6",
            "Effect": "Allow",
            "Action": [
                "ssm:GetOpsSummary"
            ],
            "Resource": [
                "*"
            ]
        }
    ]
}
```

Le rôle comprend l'entité d'approbation suivante.

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

Pour exporter OpsData depuis Explorer

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.

2. Dans le panneau de navigation, sélectionnez Explorer.
3. Choisissez Exporter le tableau.

> **Note**
>
> Lorsque vous exportez OpsData pour la première fois, le système crée un rôle d'assume pour l'exportation. Vous ne pouvez pas modifier le rôle de responsable par défaut.

4. Pour Nom du compartiment Amazon S3, choisissez un compartiment existant. Vous pouvez choisir Créer pour créer un compartiment Amazon S3 si nécessaire. Si vous ne pouvez pas modifier le nom du compartiment S3, cela signifie que vous l'avez configuré depuis la page Paramètres. Vous ne pouvez modifier le nom du compartiment que depuis la page Paramètres.

> **Note**
>
> Vous ne pouvez modifier le nom du compartiment Amazon S3 et l'ARN de la rubrique Amazon SNS qu'à partir de la page sur laquelle vous avez configuré ces paramètres pour la première fois dans Explorer.

5. Pour ARN de rubrique SNS, choisissez un ARN de rubrique Amazon SNS existant pour être averti lorsque le téléchargement est terminé.

   Si vous ne pouvez pas modifier l'ARN de la rubrique Amazon SNS, cela signifie que vous avez configuré l'ARN de la rubrique Amazon SNS depuis la page Paramètres. Vous ne pouvez modifier l'ARN de la rubrique que depuis la page Paramètres.

6. (Facultatif) Dans Message de réussite SNS, spécifiez le message de réussite que vous souhaitez afficher lorsque l'exportation est terminée avec succès.
7. Sélectionnez Envoyer. Le système accède à la page précédente et affiche le message Cliquer pour afficher le statut du processus d'exportation. Afficher les détails.

   Vous pouvez choisir Afficher les détails pour consulter le statut du runbook et la progression dans l'automatisation Systems Manager.

Vous pouvez désormais Explorer exporter OpsData depuis le compartiment Amazon S3 spécifié.

Si cette procédure ne vous permet pas d'exporter des données, vérifiez que votre utilisateur, groupe ou rôle comprend les actions `iam:CreatePolicyVersion` et `iam:DeletePolicyVersion`. Pour obtenir des informations sur l'ajout de ces actions à votre utilisateur, groupe ou rôle, veuillez consultez Modification de politiques IAM dans le Guide de l'utilisateur IAM.

Résolution des problèmes liés à Systems Manager Explorer

Cette rubrique contient des informations sur la façon de résoudre les problèmes courants avec AWS Systems Manager Explorer.

Impossible de filtrer les ressources AWS dans Explorer après une mise à jour des balises sur la page Settings (Paramètres)

Si vous mettez à jour des clés de balise ou d'autres paramètres de données dans Explorer, six heures peuvent s'écouler avant que le système ne synchronise les données en fonction de vos modifications.

Les options AWS Organizations de la page Créer une synchronisation des données de ressource sont grisées

Les options Inclure tous les comptes de ma configuration AWS Organizations et Sélectionner des unités organisationnelles dans AWS Organizations sur la page Création d'une synchronisation de données de ressources sont disponibles uniquement si vous avez paramétré et configuré AWS Organizations. Si vous installez et configurez AWS Organizations, le compte de gestion AWS Organizations ou un administrateur Explorer délégué peut créer des synchronisations de données de ressource qui utilisent ces options.

Pour plus d'informations, consultez Configuration de Systems Manager Explorer de sorte à afficher les données de plusieurs comptes et Régions et Configuration d'un administrateur délégué.

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, Explorer n'affiche pas les OpsData ni les OpsItems pour les comptes et régions dans lesquels vous n'avez pas effectué la configuration intégrée. Pour de plus amples informations, veuillez consulter Mise en route avec Systems Manager Explorer et OpsCenter.
- Lorsque vous utilisez Explorer pour afficher les données de plusieurs comptes et régions, vérifiez que vous êtes connecté au compte de gestion AWS Organizations. Pour afficher les OpsData et OpsItems de plusieurs comptes et régions, vous devez être connecté à ce compte.

Les widgets concernant des instances Amazon EC2 n'affichent pas de données

Si les widgets concernant les instances Amazon Elastic Compute Cloud (Amazon 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. Les OpsData peut prendre plusieurs minutes pour s'afficher dans Explorer après la fin de 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 source OpsData Amazon EC2 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. Cela peut prendre jusqu'à 6 heures à Systems Manager pour afficher les données de AWS Config dans les widgets EC2 d'Explorer après que vous avez initialement activé l'enregistreur de configuration ou apporté 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 appropriée dans laquelle vous avez configuré vos instances Amazon 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 au compte de gestion Organizations.

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 posséder des instances non conformes, vérifiez que vous avez installé et configuré l'application de correctifs de Systems Manager et utilisez AWS Systems Manager Patch Manager pour vérifier la conformité de votre correctif. Pour de plus amples informations, veuillez consulter AWS Systems Manager Patch Manager.

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, une fonctionnalité de AWS Systems Manager, fournit un emplacement central où les ingénieurs d'exploitation et les professionnels de l'informatique peuvent gérer les éléments de travail opérationnels (OpsItems) liés aux AWS ressources. Un OpsItem est un problème ou une interruption d'ordre opérationnel qui nécessite un examen et une correction. OpsCenter vous permet de consulter les données d'examen contextuel concernant chaque OpsItem, y compris les OpsItems et les ressources connexes. Vous pouvez également exécuter des runbooks Systems Manager Automation pour résoudre des OpsItems.

Chacun OpsItem inclut les informations pertinentes, telles que le nom et l'ID de la AWS ressource qui l'a généréOpsItem, nécessaires pour résoudre un événement. Lorsque vous le configurez OpsCenter et que vous l'intégrez à d'autres Services AWS, il peut être créé OpsItems automatiquement. S'il est intégré à ces services, OpsCenter affiche des informations provenant de AWS Config AWS CloudTrail, et Amazon EventBridge pour vous aider à étudier unOpsItem. Cela vous évite d'avoir à naviguer sur plusieurs pages de la console pendant votre examen.

Vous pouvez utiliser OpsCenter pour examiner et résoudre les problèmes liés aux nœuds gérés 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, consultez Utilisation de Systems Manager dans des environnements hybrides et multicloud.

Vous pouvez travailler avec en OpsCenter utilisant la console Systems Manager AWS Command Line Interface (AWS CLI) ou le AWS SDK de votre choix. AWS Tools for PowerShellÀ l'aide des politiques AWS Identity and Access Management (IAM), vous pouvez décider quels membres de votre organisation peuvent créer, consulter, répertorier et mettre à jourOpsItems. Vous pouvez affecter des balises aux OpsItems, puis créer des politiques IAM qui donnent accès à des utilisateurs et des groupes en fonction des balises.

> **Note**
>
> L'utilisation d'OpsCenter entraîne des frais. Pour de plus amples informations, consultez Tarification AWS Systems Manager.

Vous pouvez afficher des quotas pour toutes les fonctionnalités de Systems Manager dans la rubrique Quotas de service Systems Manager dans la Référence générale d'Amazon Web Services. Sauf indication contraire, chaque quota est spécifique à la région.

Flux de travail dans OpsCenter

Pour configurer et utiliser OpsCenter en vue de corriger des OpsItems, procédez comme suit :

1. Configurez OpsCenter. Vous pouvez également configurer OpsCenter pour la gestion centralisée des OpsItems sur l'ensemble des comptes.
2. Intégrez OpsCenter avec d'autres Services AWS. OpsCenterpeut s'intégrer à Amazon CloudWatch, Amazon CloudWatch Application Insights, Amazon EventBridge, Amazon DevOps Guru AWS Config, AWS Security Hub, et AWS Systems Manager Incident Manager.
3. Créez vos OpsItems. Vous pouvez créer des OpsItems automatiquement et manuellement.
4. Gérez les OpsItems en ajoutant du contexte aux ressources et OpsItems connexes ainsi qu'aux données opérationnelles, et en supprimant les OpsItems en double.
5. Résolvez les OpsItems à l'aide des runbooks Systems Manager Automation.

Configurer OpsCenter

AWS Systems Manager utilise une expérience de configuration intégrée pour vous aider à démarrer avec OpsCenter et Explorer quelles sont les fonctionnalités de Systems Manager. Explorerest un tableau de bord des opérations personnalisable qui fournit des informations sur vos AWS ressources. Dans cette documentation, l'installation d'Explorer et d'OpsCenter est désignée Installation intégrée.

Vous devez utiliser l'installation intégrée pour configurer OpsCenter avec Explorer. La configuration intégrée n'est disponible que dans la AWS Systems Manager console. Vous ne pouvez pas configurer Explorer ou OpsCenter par programmation. Pour plus d'informations, consultez Mise en route avec Systems Manager Explorer et OpsCenter.

Règles par défaut activées par la configuration

Lorsque vous configurezOpsCenter, vous activez les règles par défaut sur Amazon EventBridge qui se créent automatiquementOpsItems. Le tableau suivant décrit les EventBridge règles par défaut créées automatiquementOpsItems. Vous pouvez désactiver EventBridge les règles dans la page OpsCenter Paramètres, sous OpsItemRègles.

> **Important**
>
> Votre compte est débité pour les OpsItems créés par les règles par défaut. Pour plus d'informations, consultez Tarification d'AWS Systems Manager.

Nom de la règle	Description
SSMOpsItems-Autoscaling-instance-launch-failure	Cette règle crée des OpsItems lorsque le lancement d'une instance EC2 Auto Scaling a échoué.
SSMOpsItems-Autoscaling-instance-termination-failure	Cette règle crée des OpsItems lorsque la résiliation d'une instance EC2 Auto Scaling a échoué.
SSMOpsItems-EBS-snapshot-copy-failed	Cette règle crée des OpsItems lorsque le système n'a pas réussi à copier un instantané Amazon Elastic Block Store (Amazon EBS).
SSMOpsItems-EBS-snapshot-creation-failed	Cette règle crée des OpsItems lorsque le système n'a pas réussi à créer un instantané Amazon EBS.
SSMOpsItems-EBS-volume-performance-issue	Cette règle correspond à une règle AWS Health de suivi. La règle crée des OpsItems en cas de problèmes de performance avec un volume Amazon EBS (événement d'état = `AWS_EBS_DEGRADED_EBS_VOLUME_PERFORMANCE`).
SSMOpsItems-EC2-issue	Cette règle correspond à une règle AWS Health de suivi des événements inattendus qui affectent les AWS services ou les ressources. La règle crée des OpsItems lorsque, par exemple, un service envoie des communications concernant des problèmes opérationnels à l'origine de la dégradation du service ou pour attirer l'attention sur des problèmes localisés

Nom de la règle	Description
	au niveau des ressources. Par exemple, cette règle crée un OpsItem pour l'événement suivant :`AWS_EC2_OPERATIONAL_ISSUE` .
SSMOpsItems-EC2-scheduled-change	Cette règle correspond à une règle AWS Health de suivi. AWS peut planifier des événement s pour vos instances, tels que le redémarra ge, l'arrêt ou le démarrage d'instances. La règle crée des OpsItems pour les événement s EC2 planifiés. Pour plus d'informations sur les événements planifiés, consultez la section Événements planifiés pour vos instances dans le guide de l'utilisateur Amazon EC2.
SSMOpsItems-RDS-issue	Cette règle correspond à une règle AWS Health de suivi des événements inattendus qui affectent les AWS services ou les ressource s. La règle crée des OpsItems lorsque, par exemple, un service envoie des communica tions concernant des problèmes opérationnels à l'origine de la dégradation du service ou pour attirer l'attention sur des problèmes localisés au niveau des ressources. Par exemple, cette règle crée un OpsItem pour les événement s suivants : `AWS_RDS_MYSQL_DATA BASE_CRASHING_REPEATEDLY` , `AWS_RDS_EXPORT_TASK_FAILED` et `AWS_RDS_CONNECTIVITY_ISSUE` .

Nom de la règle	Description
SSMOpsItems-RDS-scheduled-change	Cette règle correspond à une règle AWS Health de suivi. La règle crée des OpsItems pour les événements Amazon RDS planifiés . Les événements planifiés fournissent des informations sur les modifications à venir pour vos ressources Amazon RDS. Certains événements peuvent vous recommander de prendre des mesures pour éviter les interruptions de service. D'autres événement s se produisent automatiquement sans action de votre part. Il est possible que votre ressource soit temporairement indisponible pendant l'activité de modification planifiée . Par exemple, cette règle crée un OpsItem pour les événements suivants : `AWS_RDS_S YSTEM_UPGRADE_SCHEDULED` et `AWS_RDS_MAINTENANCE_SCHEDULED` . Pour plus d'informations sur les événement s planifiés, consultez Catégories de types d'événements (français non garanti) dans le Guide de l'utilisateur AWS Health (français non garanti).
SSMOpsItems-SSM-maintenance-window-e xecution-failed	Cette règle crée des OpsItems lorsque le traitement de la fenêtre de maintenance Systems Manager a échoué.
SSMOpsItems-SSM-maintenance-window-e xecution-timedout	Cette règle crée des OpsItems lorsque le lancement de la fenêtre de maintenance Systems Manager expire.

Configuration de OpsCenter

Suivez la procédure ci-dessous pour configurer OpsCenter.

Pour configurer OpsCenter

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez OpsCenter.
3. Sur la page d'accueil OpsCenter, choisissez Mise en route.
4. Sur la page de OpsCenter configuration, choisissez Activer cette option pour créer automatiquement des CloudWatch événements de Explorer configuration AWS Config et Amazon en OpsItems fonction des règles et événements couramment utilisés. Si vous ne choisissez pas cette option, OpsCenter reste désactivé.

   > **Note**
   >
   > Amazon EventBridge (anciennement Amazon CloudWatch Events) fournit toutes les fonctionnalités d' CloudWatch Events et certaines nouvelles fonctionnalités, telles que des bus d'événements personnalisés, des sources d'événements tierces et un registre de schémas.

5. Sélectionnez ActiverOpsCenter.

Après avoir activé OpsCenter, vous pouvez effectuer les opérations suivantes depuis Paramètres :

- Créez des CloudWatch alarmes à l'aide du bouton Ouvrir CloudWatch la console. Pour plus d'informations, consultez Configurer les alarmes CloudWatch pour créer des OpsItems.
- Activer les informations opérationnelles. Pour plus d'informations, consultez Analyse des informations opérationnelles pour réduire OpsItems.
- Activez AWS Security Hub les alarmes liées aux résultats. Pour plus d'informations, consultez AWS Security Hub.

Table des matières

(Facultatif) Configuration d'OpsCenter pour gérer de manière centralisée les OpsItems de l'ensemble des comptes

Vous pouvez utiliser Systems Manager OpsCenter pour gérer de manière centralisée OpsItems sur plusieurs Comptes AWS dans une Région AWS sélectionnée. Cette fonction est disponible après avoir configuré votre organisation dans AWS Organizations. AWS Organizations est un service de gestion de comptes qui vous permet de consolider plusieurs comptes AWS en une organisation que vous créez et gérez de façon centralisée. AWS Organizations inclut toutes les fonctionnalités de facturation consolidée et de gestion de comptes, qui vous permettent de mieux répondre aux besoins budgétaires, de sécurité et de conformité de votre entreprise. Pour plus d'informations, consultez Présentation d'AWS Organizations dans le Guide de l'utilisateur AWS Organizations

Les utilisateurs qui appartiennent au compte de gestion AWS Organizations peuvent configurer un compte administrateur délégué pour Systems Manager. Dans le contexte de OpsCenter, les administrateurs délégués peuvent créer, modifier et afficher OpsItems dans les comptes membres. L'administrateur délégué peut également utiliser les runbooks d'automatisation de Systems Manager pour résoudre en bloc les OpsItems ou remédier aux problèmes liés aux ressources AWS qui génèrent des OpsItems.

> **Note**
>
> Vous ne pouvez désigner qu'un seul compte comme administrateur délégué pour Systems Manager. Pour de plus amples informations, veuillez consulter Création d'un administrateur AWS Organizations délégué pour Systems Manager.

Systems Manager propose les méthodes suivantes pour configurer OpsCenter afin de gérer de manière centralisée OpsItems sur plusieurs Comptes AWS.

- Quick Setup : Quick Setup, une fonctionnalité de Systems Manager, simplifie les tâches d'installation et de configuration des fonctionnalités de Systems Manager. Pour de plus amples informations, veuillez consulter AWS Systems Manager Quick Setup.

  Quick Setup pour OpsCenter vous aide à effectuer les tâches suivantes pour la gestion de OpsItems sur plusieurs comptes :

  - Enregistrement d'un compte en tant qu'administrateur délégué (si l'administrateur délégué n'a pas encore été désigné)
  - Création des politiques et des rôles AWS Identity and Access Management (IAM) requis

- Spécification d'une organisation AWS Organizations ou d'unités organisationnelles (UO) où un administrateur délégué peut gérer OpsItems sur plusieurs comptes

Pour de plus amples informations, veuillez consulter (Facultatif) Configurer OpsCenter pour gérer OpsItems sur plusieurs comptes à l'aide de Quick Setup.

> **Note**
>
> Quick Setup n'est pas disponible dans tous les Régions AWS où Systems Manager est actuellement disponible. Si Quick Setup n'est pas disponible dans une région où vous voulez l'utiliser pour configurer OpsCenter afin de gérer de manière centralisée OpsItems sur plusieurs comptes, vous devez alors utiliser la méthode manuelle. Pour consulter la liste des Régions AWS où Quick Setup est disponible, reportez-vous à Disponibilité de Quick Setup dans les Régions AWS.

- Configuration manuelle : Si Quick Setup n'est pas disponible dans la région où vous voulez configurer OpsCenter pour gérer de manière centralisée OpsItems sur plusieurs comptes, vous pouvez utiliser la procédure manuelle pour le faire. Pour de plus amples informations, veuillez consulter (Facultatif) Configuration d'OpsCenter pour gérer de manière centralisée les OpsItems de l'ensemble des comptes.

(Facultatif) Configurer OpsCenter pour gérer OpsItems sur plusieurs comptes à l'aide de Quick Setup

Quick Setup, une fonctionnalité de AWS Systems Manager, simplifie les tâches d'installation et de configuration des fonctionnalités de Systems Manager. Quick Setuppour vous OpsCenter aide à effectuer les tâches suivantes relatives à la OpsItems gestion entre comptes :

- Spécification du compte administrateur délégué
- Création de politiques et de rôles requis AWS Identity and Access Management (IAM)
- Spécification d'une AWS Organizations organisation, ou d'un sous-ensemble de comptes de membres, où un administrateur délégué peut gérer OpsItems plusieurs comptes

Lorsque vous configurez OpsCenter pour gérer OpsItems sur plusieurs comptes à l'aide de Quick Setup, Quick Setup crée les ressources suivantes dans les comptes spécifiés. Ces ressources autorisent les comptes spécifiés à travailler avec OpsItems et à utiliser des runbooks d'automatisation pour résoudre les problèmes liés à la génération OpsItems de AWS ressources.

Ressources	Comptes
Rôle lié au service AWS Identity and Access Management (IAM) `AWSServiceRoleForAmazonSSM_AccountDiscovery` Pour plus d'informations sur ce rôle, consultez Utilisation des rôles pour collecter des Compte AWS informations pour OpsCenter et Explorer.	AWS Organizations compte de gestion et compte administrateur délégué
Rôle IAM `OpsItem-CrossAccountManagementRole` Rôle IAM `AWS-SystemsManager-AutomationAdministrationRole`	Compte administrateur délégué
Rôle IAM `OpsItem-CrossAccountExecutionRole` Rôle IAM `AWS-SystemsManager-AutomationExecutionRole` Politique de ressources de Systems Manager `AWS::SSM::ResourcePolicy` pour le groupe par défaut OpsItem (`OpsItemGroup`)	Tous les comptes des AWS Organizations membres

> **Note**
>
> Si vous avez déjà configuré OpsCenter pour gérer OpsItems plusieurs comptes à l'aide de la méthode manuelle, vous devez supprimer les AWS CloudFormation piles ou les ensembles de piles créés au cours des étapes 4 et 5 de ce processus. Si ces ressources existent dans votre compte lorsque vous effectuez la procédure suivante, Quick Setup ne configure pas correctement la gestion d'OpsItem sur plusieurs comptes.

Pour configurer OpsCenter afin de gérer OpsItems sur plusieurs comptes à l'aide de Quick Setup

1. Connectez-vous à l' AWS Management Console aide du compte AWS Organizations de gestion.
2. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
3. Dans le panneau de navigation, sélectionnez Quick Setup.
4. Choisissez l'onglet Bibliothèque.
5. Faites défiler la page vers le bas et localisez la vignette de configuration OpsCenter. Choisissez Créer.
6. Sur la page Quick Setup OpsCenter, dans la section Administrateur délégué, saisissez un ID de compte. Si vous ne parvenez pas à modifier ce champ, cela signifie qu'un compte d'administrateur délégué a déjà été spécifié pour Systems Manager.
7. Dans la section Targets (Cibles), sélectionnez une option. Si vous choisissez Personnalisé, sélectionnez les unités organisationnelles (UO) dans lesquelles vous voulez gérer OpsItems sur plusieurs comptes.
8. Choisissez Créer.

Quick Setup crée la configuration OpsCenter et déploie les ressources AWS requises vers les UO désignées.

> **Note**
>
> Si vous ne voulez pas gérer OpsItems sur plusieurs comptes, vous pouvez supprimer la configuration de Quick Setup. Lorsque vous supprimez la configuration, Quick Setup supprime les politiques et rôles IAM suivants créés lors du déploiement initial de la configuration :
>
> - `OpsItem-CrossAccountManagementRole` du compte de l'administrateur délégué
> - `OpsItem-CrossAccountExecutionRole` et `SSM::ResourcePolicy` de tous les comptes membres de l'organisation
>
> Quick Setup supprime la configuration de toutes les unités organisationnelles et Régions AWS où la configuration a été initialement déployée.

Résolution des problèmes liés à une configuration Quick Setup pour OpsCenter

Cette section contient des informations qui vous aideront à résoudre les problèmes rencontrés lors de la configuration de la gestion OpsItem multicomptes en utilisant Quick Setup.

Rubriques

- Le déploiement vers ceux-ci StackSets a échoué : DelegatedAdmin
- L'état de la configuration Quick Setup indique Échec

Le déploiement vers ceux-ci StackSets a échoué : DelegatedAdmin

Lors de la création d'une configuration OpsCenter, Quick Setup déploie deux jeux de piles AWS CloudFormation dans le compte de gestion d'Organizations. Les jeux de piles utilisent le préfixe suivant : `AWS-QuickSetup-SSMOpsCenter`. Si Quick Setup affiche l'erreur suivante : `Deployment to these StackSets failed: delegatedAdmin` utilise la procédure suivante pour résoudre ce problème.

Pour résoudre une erreur StackSets failed:DelegatedAdmin

1. Si vous avez reçu le `Deployment to these StackSets failed: delegatedAdmin` message d'erreur sous forme de bannière rouge Quick Setup sur la console, connectez-vous au compte d'administrateur délégué et à la région d'Quick Setuporigine Région AWS désignée.
2. Ouvrez la AWS CloudFormation console à l'adresse https://console.aws.amazon.com/cloudformation.
3. Sélectionnez la pile créée par votre configuration Quick Setup. Le nom de la pile inclut les éléments suivants : AWS- QuickSetup -SSM OpsCenter.

   > **Note**
   >
   > CloudFormation Supprime parfois les déploiements de stack ayant échoué. Si la pile n'est pas disponible dans le tableau Stacks (Piles), sélectionnez Deleted (Supprimé) dans la liste des filtres.

4. Affichez les éléments Status (Statut) et Status reason (Raison du statut). Pour plus d'informations sur les statuts de pile, consultez Codes de statut de la pile dans le Guide de l'utilisateur AWS CloudFormation .

5. Pour comprendre l'étape exacte qui a échoué, consultez l'onglet Events (Événements) et passez en revue chaque Status (Statut) d'événement. Pour plus d'informations, consultez Résolution des problèmes dans le Guide de l'utilisateur AWS CloudFormation .

> **Note**
>
> Si vous ne parvenez pas à résoudre l'échec du déploiement à l'aide des étapes de CloudFormation dépannage, supprimez la configuration et réessayez.

L'état de la configuration Quick Setup indique Échec

Si le tableau des détails de configuration de la page des détails de la configuration indique un statut de configuration de`Failed`, connectez-vous à la région Compte AWS et dans laquelle elle a échoué.

Pour résoudre un échec de Quick Setup à créer une configuration OpsCenter

1. Connectez-vous au Compte AWS et à l' Région AWS endroit où la panne s'est produite.
2. Ouvrez la AWS CloudFormation console à l'adresse https://console.aws.amazon.com/cloudformation.
3. Sélectionnez la pile créée par votre configuration Quick Setup. Le nom de la pile inclut les éléments suivants : AWS- QuickSetup -SSM OpsCenter.

   > **Note**
   >
   > CloudFormation Supprime parfois les déploiements de stack ayant échoué. Si la pile n'est pas disponible dans le tableau Stacks (Piles), sélectionnez Deleted (Supprimé) dans la liste des filtres.

4. Affichez les éléments Status (Statut) et Status reason (Raison du statut). Pour plus d'informations sur les statuts de pile, consultez Codes de statut de la pile dans le Guide de l'utilisateur AWS CloudFormation .
5. Pour comprendre l'étape exacte qui a échoué, consultez l'onglet Events (Événements) et passez en revue chaque Status (Statut) d'événement. Pour plus d'informations, consultez Résolution des problèmes dans le Guide de l'utilisateur AWS CloudFormation .

La configuration du compte membre indique ResourcePolicyLimitExceededException

Si l'état d'une pile affiche `ResourcePolicyLimitExceededException`, le compte a précédemment été intégré à la gestion inter-comptes OpsCenter à l'aide de la méthode manuelle. Pour résoudre ce problème, vous devez supprimer les AWS CloudFormation piles ou les ensembles de piles créés au cours des étapes 4 et 5 du processus d'intégration manuel. Pour plus d'informations, voir Supprimer un ensemble de piles et Supprimer une pile sur la AWS CloudFormation console dans le Guide de AWS CloudFormation l'utilisateur.

(Facultatif) Configuration d'OpsCenter pour gérer de manière centralisée les OpsItems de l'ensemble des comptes

Cette section décrit comment configurer manuellement OpsCenter pour la gestion OpsItem multicomptes. Bien que ce processus soit toujours pris en charge, il a été remplacé par un processus plus récent qui utilise Systems Manager Quick Setup. Pour plus d'informations, consultez (Facultatif) Configurer OpsCenter pour gérer OpsItems sur plusieurs comptes à l'aide de Quick Setup.

Vous pouvez configurer un compte central pour créer manuellement des OpsItems pour les comptes membre, et gérer et corriger ces OpsItems. Le compte central peut être le compte de AWS Organizations gestion, ou à la fois le compte AWS Organizations de gestion et le compte d'administrateur délégué de Systems Manager. Nous vous recommandons d'utiliser le compte d'administrateur délégué de Systems Manager comme compte central. Vous ne pouvez utiliser cette fonction qu'après avoir configuré AWS Organizations.

Vous pouvez ainsi en consolider plusieurs au Comptes AWS sein d'une organisation que vous créez et gérez de manière centralisée. AWS Organizations L'utilisateur du compte central peut créer simultanément des OpsItems pour les comptes de tous les membres sélectionnés et gérer ces OpsItems.

Utilisez le processus décrit dans cette section pour activer le principal du service Systems Manager dans Organizations et configurer les autorisations AWS Identity and Access Management (IAM) pour travailler avec OpsItems plusieurs comptes.

Rubriques

- Étape 5 : Configuration des autorisations pour utiliser des ressources connexes sur plusieurs comptes

> **Note**
>
> Seuls les OpsItems de type `/aws/issue` sont pris en charge lors de l'utilisation d'OpsCenter sur plusieurs comptes.

Avant de commencer

Avant de configurer OpsCenter pour travailler sur lesOpsItems de plusieurs comptes, assurez-vous d'avoir configuré les éléments suivants :

- Un compte administrateur délégué pour Systems Manager Pour plus d'informations, consultez Configuration d'un administrateur délégué.
- Une organisation définie et configurée dans Organizations. Pour plus d'informations, consultez la rubrique Création et gestion d'une organisation dans le Guide de l'utilisateur AWS Organizations .
- Vous avez configuré Systems Manager Automation pour exécuter des runbooks d'automatisation sur plusieurs Régions AWS AWS comptes. Pour plus d'informations, consultez Exécution d'automatisations dans plusieurs régions et comptes Régions AWS.

Étapo 1 : Création d'une synchronisation des données de ressource

Après avoir configuré et configuré AWS Organizations, vous pouvez les agréger OpsItems OpsCenter pour l'ensemble d'une organisation en créant une synchronisation des données de ressources. Pour plus d'informations, consultez Création d'une synchronisation des données de ressource. Lorsque vous créez la synchronisation, dans la section Ajouter des comptes, assurez-vous de choisir l'option Inclure tous les comptes depuis ma AWS Organizations configuration.

Étape 2 : Activation du principal de service Systems Manager dans AWS Organizations

Pour permettre à un utilisateur de travailler avec OpsItems plusieurs comptes, le principal de service Systems Manager doit être activé dans AWS Organizations. Si vous avez déjà configuré Systems Manager pour des scénarios multicomptes à l'aide d'autres fonctionnalités, le principal du service Systems Manager est peut-être déjà configuré dans Organizations. Exécutez les commandes suivantes à partir du AWS Command Line Interface (AWS CLI) pour vérifier. Si vous n'avez pas

configuré Systems Manager pour d'autres scénarios multicomptes, passez à la procédure suivante, à savoir Activer le principal de service Systems Manager dans AWS Organizations.

Pour vérifier que le principal de service Systems Manager est activé dans AWS Organizations

1. Téléchargez la dernière version du sur AWS CLI votre ordinateur local.
2. Ouvrez le AWS CLI, et exécutez la commande suivante pour spécifier vos informations d'identification et un Région AWS.

```
aws configure
```

Le système vous invite à spécifier les informations suivantes. Dans les exemples suivants, remplacez chaque *espace réservé pour l'entrée utilisateur* par vos propres informations.

```
AWS Access Key ID [None]: key_name
AWS Secret Access Key [None]: key_name
Default region name [None]: region
Default output format [None]: ENTER
```

3. Exécutez la commande suivante pour vérifier que le principal de service Systems Manager est activé pour AWS Organizations.

```
aws organizations list-aws-service-access-for-organization
```

La commande renvoie des informations semblables à celles de l'exemple suivant.

```
{
    "EnabledServicePrincipals": [
        {
            "ServicePrincipal":
 "member.org.stacksets.cloudformation.amazonaws.com",
            "DateEnabled": "2020-12-11T16:32:27.732000-08:00"
        },
        {
            "ServicePrincipal": "opsdatasync.ssm.amazonaws.com",
            "DateEnabled": "2022-01-19T12:30:48.352000-08:00"
        },
        {
            "ServicePrincipal": "ssm.amazonaws.com",
```

```
            "DateEnabled": "2020-12-11T16:32:26.599000-08:00"
        }
    ]
}
```

Pour activer le principal de service Systems Manager dans AWS Organizations

Si vous n'avez pas encore configuré le principal de service Systems Manager pour Organizations, procédez comme suit. Pour plus d'informations sur cette commande, reportez-vous enable-aws-service-accessà la référence des AWS CLI commandes.

1. Installez et configurez le AWS Command Line Interface (AWS CLI), si ce n'est pas déjà fait. Pour de plus amples informations, consultez Installation d' et Configuration de l'.
2. Téléchargez la dernière version du sur AWS CLI votre ordinateur local.
3. Ouvrez le AWS CLI, et exécutez la commande suivante pour spécifier vos informations d'identification et un Région AWS.

   ```
   aws configure
   ```

   Le système vous invite à spécifier les informations suivantes. Dans les exemples suivants, remplacez chaque *espace réservé pour l'entrée utilisateur* par vos propres informations.

   ```
   AWS Access Key ID [None]: key_name
   AWS Secret Access Key [None]: key_name
   Default region name [None]: region
   Default output format [None]: ENTER
   ```
4. Exécutez la commande suivante pour activer le principal de service Systems Manager pour AWS Organizations.

   ```
   aws organizations enable-aws-service-access --service-principal "ssm.amazonaws.com"
   ```

Étape 3 : Création du rôle lié au service **AWSServiceRoleForAmazonSSM_AccountDiscovery**

Un rôle lié à un service tel que le AWSServiceRoleForAmazonSSM_AccountDiscovery rôle est un type unique de rôle IAM directement lié à un Service AWS, tel que Systems Manager. Les rôles liés au service sont prédéfinis par le service et incluent toutes les autorisations dont le service

a besoin pour appeler d'autres personnes en votre Services AWS nom. Pour de plus amples informations sur le rôle lié à un service `AWSServiceRoleForAmazonSSM_AccountDiscovery`, consultez Autorisations des rôles liés à un service pour la découverte de comptes Systems Manager.

Utilisez la procédure suivante pour créer le rôle lié à service `AWSServiceRoleForAmazonSSM_AccountDiscovery` à l'aide de l' AWS CLI. Pour plus d'informations sur la commande utilisée dans cette procédure, reportez-vous create-service-linked-roleà la référence des AWS CLI commandes.

Créer le rôle lié à un service **`AWSServiceRoleForAmazonSSM_AccountDiscovery`**

1. Connectez-vous au compte de AWS Organizations gestion.
2. Lorsque vous êtes connecté au compte de gestion d'Organizations, exécutez la commande suivante.

```
aws iam create-service-linked-role \
    --aws-service-name accountdiscovery.ssm.amazonaws.com \
    --description "Systems Manager account discovery for AWS Organizations service-
linked role"
```

Étape 4 : Configuration des autorisations pour travailler sur les OpsItems de plusieurs comptes

Utilisez des AWS CloudFormation stacksets pour créer une politique de `OpsItemGroup` ressources et un rôle d'exécution IAM qui autorisent les utilisateurs à travailler avec OpsItems plusieurs comptes. Pour commencer, téléchargez et décompressez le fichier `OpsCenterCrossAccountMembers.zip`. Ce fichier contient le fichier `OpsCenterCrossAccountMembers.yaml` AWS CloudFormation modèle. Lorsque vous créez un ensemble de piles à l'aide de ce modèle, la politique de `OpsItemCrossAccountResourcePolicy` ressources et le rôle `OpsItemCrossAccountExecutionRole` d'exécution sont CloudFormation automatiquement créés dans le compte. Pour plus d'informations sur la création de jeux de piles, consultez la rubrique Créer un ensemble de piles dans le Guide de l'utilisateur AWS CloudFormation .

> **Important**
>
> Veuillez tenir compte des informations importantes suivantes relatives à cette tâche :
>
> - Vous devez déployer l'ensemble de piles tout en étant connecté au compte de gestion AWS Organizations .

- Vous devez répéter cette procédure lorsque vous êtes connecté à chaque compte que vous souhaitez cibler pour utiliser des OpsItems sur plusieurs comptes, y compris le compte d'administrateur délégué.
- Si vous souhaitez activer OpsItems l'administration entre comptes dans d'autres applications Régions AWS, choisissez Ajouter toutes les régions dans la section Spécifier les régions du modèle. L'administration intercompte OpsItem n'est pas prise en charge dans les régions d'adhésion.

Étape 5 : Configuration des autorisations pour utiliser des ressources connexes sur plusieurs comptes

Un OpsItem peut comprendre des informations détaillées sur les ressources affectées, comme les instances Amazon Elastic Compute Cloud (Amazon EC2) ou les compartiments Amazon Simple Storage Service (Amazon S3). Le rôle d'exécution `OpsItemCrossAccountExecutionRole`, que vous avez créé lors de l'étape précédente, fournit à OpsCenter des autorisations en lecture seule permettant aux comptes membre de consulter les ressources connexes. Vous devez également créer un rôle IAM pour autoriser les comptes de gestion à consulter les ressources connexes et à interagir avec elles, ce que vous effectuerez au cours de cette tâche.

Pour commencer, téléchargez et décompressez le fichier `OpsCenterCrossAccountManagementRole.zip`. Ce fichier contient le fichier `OpsCenterCrossAccountManagementRole.yaml` AWS CloudFormation modèle. Lorsque vous créez une pile à l'aide de ce modèle, le rôle `OpsCenterCrossAccountManagementRole` IAM est CloudFormation automatiquement créé dans le compte. Pour plus d'informations sur la création d'une pile, consultez la section Création d'une pile sur la AWS CloudFormation console dans le Guide de AWS CloudFormation l'utilisateur.

Important

Veuillez tenir compte des informations importantes suivantes relatives à cette tâche :

- Si vous envisagez de définir un compte en tant qu'administrateur délégué pourOpsCenter, veillez à le spécifier Compte AWS lors de la création de la pile.
- Vous devez effectuer cette procédure lorsque vous êtes connecté au compte de gestion AWS Organizations , puis à nouveau lorsque vous êtes connecté au compte d'administrateur délégué.

(Facultatif) Configurer Amazon SNS pour recevoir des notifications sur OpsItems

Vous pouvez configurer OpsCenter pour envoyer des notifications à une rubrique Amazon Simple Notification Service (Amazon SNS) lorsque le système crée un OpsItem ou met à jour un OpsItem existant.

Effectuez les étapes suivantes pour recevoir des notifications pour OpsItems.

- Étape 1 : Créer une rubrique Amazon SNS et s'y abonner
- Étape 2 : Mise à jour de la politique d'accès Amazon SNS
- Étape 3 : Mise à jour de la politique d'accès AWS KMS

> **Note**
>
> Si vous activez le chiffrement côté serveur AWS Key Management Service (AWS KMS) à l'étape 2, vous devez terminer l'étape 3. Sinon, vous pouvez passer à l'étape 3.

- Étape 4 : Activer les règles OpsItems par défaut pour envoyer des notifications pour les nouveaux OpsItems

Étape 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 plus d'informations, consultez les sections Créer une rubrique et Abonnement d'un point de terminaison à une rubrique Amazon SNS, dans le Guide du développeur d'Amazon Simple Notification Service.

> **Note**
>
> Si vous utilisez OpsCenter plusieurs comptes Régions AWS ou, vous devez créer une rubrique Amazon SNS et vous y abonner dans chaque région ou compte pour lequel vous souhaitez recevoir des OpsItem notifications.

Étape 2 : Mise à jour de la politique d'accès Amazon SNS

Vous devez associer une rubrique Amazon SNS aux OpsItems. Utilisez la procédure suivante pour configurer une stratégie d'accès Amazon SNS afin que Systems Manager puisse publier les notifications d'OpsItems dans la rubrique Amazon SNS créée à l'étape 1.

1. Connectez-vous à la console Amazon SNS AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/sns/v3/home.
2. Dans le volet de navigation, choisissez Rubriques.
3. Sélectionnez la rubrique que vous avez créée à l'étape 1, puis sélectionnez Modifier.
4. Développez la politique d'accès.
5. Ajoutez le bloc `Sid` suivant à la politique existante. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations.

```
{
      "Sid": "Allow OpsCenter to publish to this topic",
      "Effect": "Allow",
      "Principal": {
        "Service": "ssm.amazonaws.com"
      },
      "Action": "SNS:Publish",
      "Resource": "arn:aws:sns:region:account ID:topic name", // Account ID of the
 SNS topic owner
      "Condition": {
      "StringEquals": {
        "AWS:SourceAccount": "account ID" //  Account ID of the OpsItem owner
      }
   }
}
```

> **Note**
>
> La clé de condition globale `aws:SourceAccount` protège contre le scénario d'adjoint confus. Pour utiliser cette clé de condition, définissez la valeur sur l'ID de compte du propriétaire d'OpsItem. Pour plus d'informations, veuillez consulter la rubrique Adjoint confus dans le Guide de l'utilisateur IAM.

6. Sélectionnez 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 configurez la rubrique Amazon SNS avec une clé de chiffrement côté serveur AWS Key Management Service (AWS KMS) à l'étape 2, passez à l'étape 3. Sinon, vous pouvez passer à l'étape 3.

Étape 3 : Mise à jour de la politique d'accès AWS KMS

Si vous avez activé le chiffrement AWS KMS côté serveur pour votre rubrique Amazon SNS, vous devez également mettre à jour la politique d'accès que vous avez choisie lors de AWS KMS key la configuration de la rubrique. Utilisez la procédure suivante pour mettre à jour la politique d'accès afin que Systems Manager puisse publier les notifications d'OpsItem dans la rubrique Amazon SNS créée à l'étape 1.

> **Note**
>
> OpsCenter ne prend pas en charge la publication des OpsItems sur une rubrique Amazon SNS configurée avec une Clé gérée par AWS.

1. Ouvrez la AWS KMS console à l'adresse https://console.aws.amazon.com/kms.
2. Pour modifier le Région AWS, utilisez le sélecteur de région dans le coin supérieur droit de la page.
3. Dans le volet de navigation, sélectionnez Clés gérées par le client.
4. Sélectionnez l'ID de la clé KMS que vous avez choisie lors de la création de la rubrique.
5. Dans la section Key policy (Politique de clé), sélectionnez Switch to policy view (Passer à la vue de politique).
6. Sélectionnez Edit (Modifier).
7. Ajoutez le bloc `Sid` suivant à la politique existante. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations.

```
{
      "Sid": "Allow OpsItems to decrypt the key",
      "Effect": "Allow",
      "Principal": {
        "Service": "ssm.amazonaws.com"
```

```
      },
      "Action": ["kms:Decrypt", "kms:GenerateDataKey*"],
       "Resource": "arn:aws:kms:region:account ID:key/key ID"
    }
```

Dans l'exemple suivant, le nouveau bloc est entré à la ligne 14.

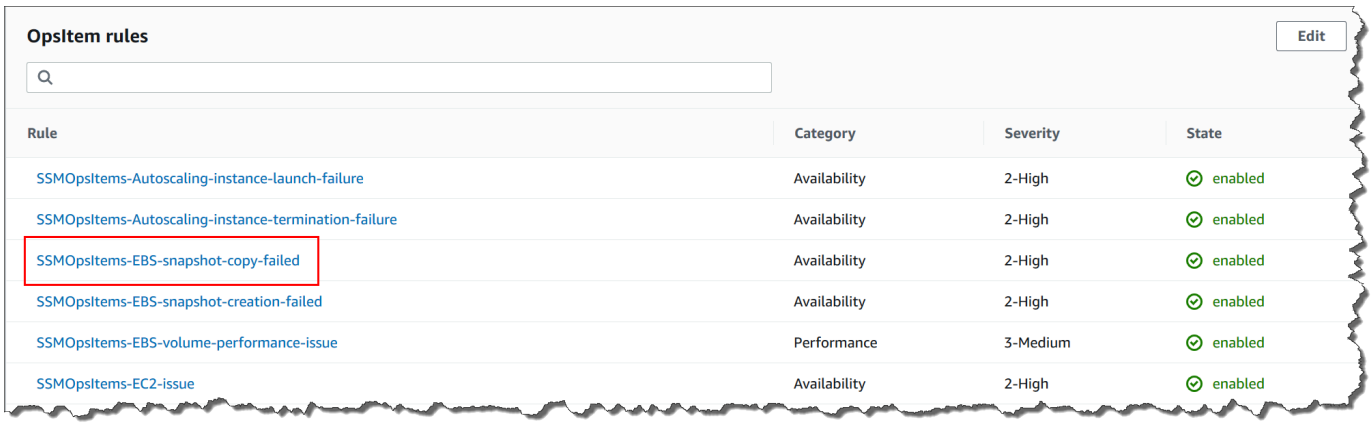
8. Sélectionnez Enregistrer les modifications.

Étape 4 : Activer les règles OpsItems par défaut pour envoyer des notifications pour les nouveaux OpsItems

OpsItemsLes règles par défaut d'Amazon EventBridge ne sont pas configurées avec un Amazon Resource Name (ARN) pour les notifications Amazon SNS. Utilisez la procédure suivante pour modifier une règle EventBridge et saisir un `notifications` bloc.

Pour ajouter un bloc de notifications à une règle OpsItem par défaut

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez OpsCenter.
3. Sélectionnez l'onglet OpsItems, puis sélectionnez Configurer les sources.
4. Sélectionnez le nom de la règle source que vous souhaitez configurer avec un bloc `notifications`, comme illustré dans l'exemple suivant.



La règle s'ouvre sur Amazon EventBridge.

5. Sur la page des détails de la règle, sur l'onglet Targets(Cibles), choisissez Edit (Modifier).
6. Dans la section Additional settings (Réglages supplémentaires), choisissez Configure input transformer (Configurer le transformateur d'entrée).
7. Dans la case Modèle, ajoutez un `notifications` au format suivant.

```
"notifications":[{"arn":"arn:aws:sns:region:account ID:topic name"}],
```

Voici un exemple :

```
"notifications":[{"arn":"arn:aws:sns:us-west-2:1234567890:MySNSTopic"}],
```

Entrez le bloc de notifications avant le `resources` bloc, comme indiqué dans l'exemple suivant pour la région USA Ouest (Oregon) (us-west-2).

```
{
    "title": "EBS snapshot copy failed",
    "description": "CloudWatch Event Rule SSMOpsItems-EBS-snapshot-copy-failed was
 triggered. Your EBS snapshot copy has failed. See below for more details.",
    "category": "Availability",
    "severity": "2",
    "source": "EC2",
    "notifications": [{
        "arn": "arn:aws:sns:us-west-2:1234567890:MySNSTopic"
    }],
    "resources": <resources>,
    "operationalData": {
        "/aws/dedup": {
```

```
            "type": "SearchableString",
            "value": "{\"dedupString\":\"SSMOpsItems-EBS-snapshot-copy-failed\"}"
        },
        "/aws/automations": {
            "value": "[ { \"automationType\": \"AWS:SSM:Automation\",
 \"automationId\": \"AWS-CopySnapshot\" } ]"
        },
        "failure-cause": {
            "value": <failure - cause>
        },
        "source": {
            "value": <source>
        },
        "start-time": {
            "value": <start - time>
        },
        "end-time": {
            "value": <end - time>
        }
    }
}
```

8. Choisissez Confirmer.
9. Choisissez Suivant.
10. Choisissez Suivant.
11. Choisissez Mettre à jour la règle.

La prochaine fois que le système crée un OpsItem pour la règle par défaut, il publie une notification dans la rubrique Amazon SNS.

Intégration des OpsCenter à d'autres Services AWS

OpsCenter, une fonctionnalité de AWS Systems Manager, s'intègre à plusieurs Services AWS pour diagnostiquer et résoudre les problèmes liés aux AWS ressources. Vous devez configurer le Service AWS avant de l'intégrer à OpsCenter.

Par défaut, les éléments suivants Services AWS sont intégrés OpsCenter et peuvent être créés OpsItems automatiquement :

- Amazon CloudWatch
- Informations sur les CloudWatch applications Amazon

- Amazon EventBridge
- AWS Config
- AWS Systems Manager Incident Manager

Vous devez intégrer les services suivants à OpsCenter pour créer des OpsItems automatiquement :

- Amazon DevOps Guru
- AWS Security Hub

Lorsque l'un de ces services crée un OpsItem, vous pouvez gérer et corriger l'OpsItem depuis OpsCenter. Pour plus d'informations, consultez Gestion des OpsItems et Correction des problèmes d'OpsItem.

Pour plus d'informations sur chacun d'entre eux Service AWS et sur la manière dont ils s'intègrentOpsCenter, consultez les rubriques suivantes.

Rubriques

- Amazon CloudWatch
- Informations sur les CloudWatch applications Amazon
- Amazon DevOps Guru
- Amazon EventBridge
- AWS Config
- AWS Security Hub
- Incident Manager

Amazon CloudWatch

Amazon CloudWatch surveille vos AWS ressources et vos services, et affiche des statistiques sur chaque service Service AWS que vous utilisez. CloudWatch crée un OpsItem lorsqu'une alarme entre dans l'état d'alarme. Par exemple, vous pouvez configurer une alarme pour créer automatiquement un OpsItem en cas de pic d'erreurs HTTP générées par votre Application Load Balancer.

Certaines alarmes que vous pouvez configurer CloudWatch pour créer OpsItems sont répertoriées dans la liste suivante :

- Amazon DynamoDB : les actions de lecture et d'écriture de la base de données atteignent un seuil

- Amazon EC2 : l'utilisation du processeur atteint un seuil
- AWS facturation : les frais estimés atteignent un seuil
- Amazon EC2 : une instance échoue à une vérification de son statut
- Amazon Elastic Block Store (EBS) : l'utilisation de l'espace disque atteint un seuil

Vous pouvez créer une alarme ou modifier une alarme existante pour créer un OpsItem. Pour plus d'informations, consultez Configurer les alarmes CloudWatch pour créer des OpsItems.

Lorsque vous activez OpsCenter l'utilisation de la configuration intégrée, celle-ci s' CloudWatch intègre àOpsCenter.

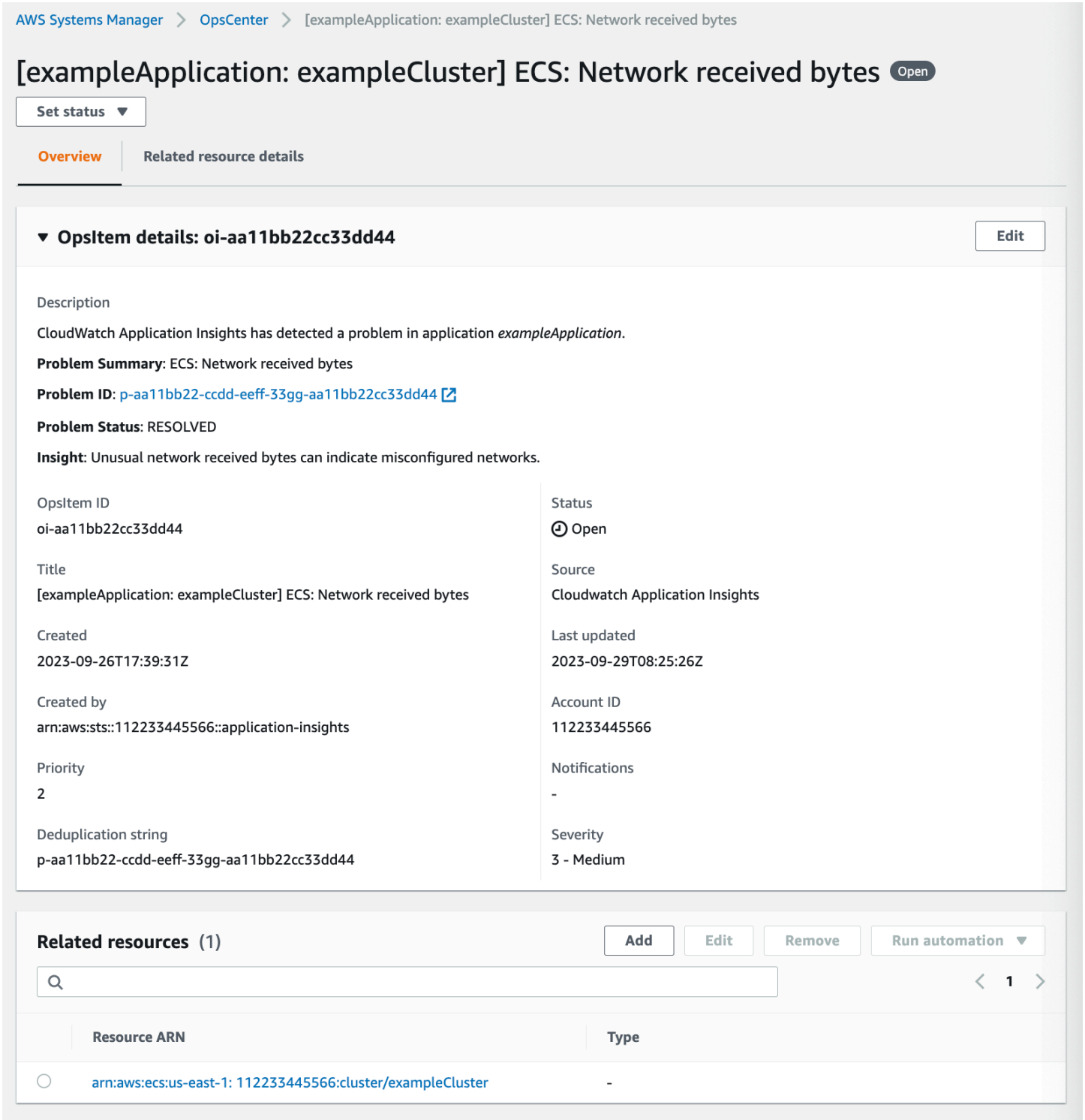
Informations sur les CloudWatch applications Amazon

À l'aide CloudWatch d'Amazon Application Insights, vous pouvez configurer les moniteurs les plus appropriés pour les ressources de vos applications afin d'analyser en permanence les données afin de détecter tout signe de problème avec vos applications. Lorsque vous configurez les ressources de CloudWatch l'application dans Application Insights, vous pouvez choisir de faire OpsItems en sorte que le système les créeOpsCenter. Un OpsItem est créé dans la console OpsCenter pour chaque problème détecté avec l'application. Pour plus d'informations, consultez la section Configurer, configurer et gérer votre application de surveillance dans le guide de CloudWatch l'utilisateur Amazon.

> **Note**
>
> À compter du 16 octobre 2023, le titre et la description de OpsItems Created by CloudWatch Application Insights utilisent désormais le format amélioré suivant :

```
OpsItem title: [<APPLICATION NAME>: <RESOURCE ID>] <PROBLEM SUMMARY>

OpsItem description:

CloudWatch Application Insights has detected a problem in application <APPLICATION
 NAME>.
Problem summary: <PROBLEM SUMMARY>
Problem ID: <PROBLEM ID> (hyperlinks to the Application Insights problem summary page)
Problem Status: <PROBLEM STATUS>
Insight: <INSIGHT>
```

Voici un exemple :



Amazon DevOps Guru

Amazon DevOps Guru utilise le machine learning pour analyser vos données opérationnelles, les métriques de vos applications et les événements liés aux applications afin d'identifier les

comportements qui s'écartent des modèles de fonctionnement normaux. Si vous permettez à DevOps Guru de générer un OpsItem inOpsCenter, chaque information en génère une nouvelleOpsItem. Vous pouvez utiliser OpsCenter pour gérer vos OpsItems.

DevOpsGuru crée automatiquementOpsItems. Vous pouvez permettre à Amazon DevOps Guru de créer OpsItems Quick Setup en utilisant une fonctionnalité de Systems Manager. Le système crée OpsItems en utilisant le rôle lié au service AWSServiceRoleForDevOpsGuru AWS Identity and Access Management (IAM).

Pour intégrer OpsCenter DevOps Guru

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Quick Setup.
3. Sur la page des options de configuration de Customize DevOps Guru, choisissez l'onglet Bibliothèque.
4. Dans le volet DevOpsGuru, choisissez Create.
5. Pour les options de configuration, sélectionnez Activer AWS Systems Manager OpsItems.
6. Sélectionnez Créer après avoir terminé la configuration.

Amazon EventBridge

Amazon EventBridge diffuse un flux d'événements décrivant les modifications apportées aux AWS ressources. Lorsque vous activez OpsCenter l'option à l'aide de la configuration intégrée, celle-ci s'intègre EventBridge aux OpsCenter EventBridge règles par défaut et les active. Sur la base de ces règles, EventBridge créeOpsItems. À l'aide de règles, vous pouvez filtrer les événements et les acheminer vers OpsCenter afin d'y mener un examen et d'appliquer des correctifs.

> **Note**
>
> Amazon EventBridge (anciennement Amazon CloudWatch Events) fournit toutes les fonctionnalités d' CloudWatch Events et certaines nouvelles fonctionnalités, telles que des bus d'événements personnalisés, des sources d'événements tierces et un registre de schémas.

Voici quelques règles que vous pouvez configurer EventBridge pour créer un OpsItem :

- Security Hub : alerte de sécurité émise
- Amazon DynamoDB : un événement de limitation
- Amazon EC2 Auto Scaling : échec du lancement d'une instance
- Systems Manager : échec de l'exécution d'une automatisation
- AWS Health : une alerte pour la maintenance planifiée
- Amazon EC2 : l'état de l'instance est passé de « en cours d'exécution » à « arrêté »

En fonction de vos besoins, vous pouvez créer une règle ou modifier une règle existante pour créer un OpsItems. Pour obtenir des instructions sur la modification d'une règle pour créer un OpsItem, consultez Configurer des règles EventBridge pour créer des OpsItems.

AWS Config

AWS Config fournit une vue détaillée de la configuration des AWS ressources de votre Compte AWS.

AWS Config ne s'intègre pas directement àOpsCenter. Au lieu de cela, vous créez une AWS Config règle qui envoie un événement à Amazon EventBridge, par exemple lorsqu'une instance non conforme est AWS Config détectée. EventBridge Évalue ensuite cet événement par rapport à une EventBridge règle que vous avez créée. Si la règle correspond, EventBridge transforme l'événement en un OpsItem et le transmet en OpsCenter tant que cible de destination.

Utiliser cet OpsItem vous permet de suivre les détails de la ressource non conforme, d'enregistrer les actions d'investigation et de donner accès à des mesures correctives cohérentes.

Informations connexes

Configurer des règles EventBridge pour créer des OpsItems

Utilisation AWS Systems ManagerOpsCenter et AWS Config pour le suivi de la conformité

AWS Security Hub

AWS Security Hub collecte des données de sécurité, appelées conclusions, provenant de l'ensemble Comptes AWS des services. En utilisant un ensemble de règles pour détecter et générer des résultats, Security Hub vous aide à identifier, prioriser et remédier aux problèmes de sécurité pour les ressources que vous gérez. Après avoir configuré l'intégration, comme décrit dans cette rubrique, Systems Manager crée des OpsItems pour les résultats de Security Hub dans OpsCenter.

> **Note**
>
> L'intégration de OpsCenter à Security Hub est bidirectionnelle. Cela signifie que si vous mettez à jour le champ Statut ou Gravité d'un OpsItem lié à un résultat de sécurité, le système synchronise les modifications avec Security Hub. De même, toute modification apportée à un résultat est automatiquement mise à jour dans les OpsItems correspondant de l'OpsCenter.
>
> Lorsqu'un OpsItem est créé à partir d'une découverte du Security Hub, les métadonnées du Security Hub sont automatiquement ajoutées au champ de données opérationnelles duOpsItem. Si ces métadonnées sont supprimées, les mises à jour bidirectionnelles ne fonctionnent plus.

Par défaut, Systems Manager crée OpsItems pour les résultats critiques et de gravité élevée. Vous pouvez configurer manuellement OpsCenter pour créer OpsItems pour les résultats de gravité moyenne et faible. OpsCenter ne crée pas d'OpsItems pour les résultats informationnels, car ils ne nécessitent pas de remédiation. Pour plus d'informations sur les niveaux de gravité de Security Hub, consultez Gravité (français non garanti) dans la Référence d'API AWS Security Hub .

Avant de commencer

Avant de configurer OpsCenter pour créer OpsItems en fonction des résultats de Security Hub, vérifiez que vous avez terminé les tâches de configuration de Security Hub. Pour de plus amples informations, veuillez consulter Configuration de Security Hub dans le Guide de l'utilisateur AWS Security Hub .

Lorsque vous intégrez Security Hub à OpsCenter, le système crée OpsItems en utilisant le rôle lié à un service IAM `AWSServiceRoleForSystemsManagerOpsDataSync`. Pour plus d'informations sur ce rôle, consultez Utiliser des rôles pour créer OpsData et OpsItems pour Explorer.

> **Warning**
>
> Notez les informations importantes suivantes concernant la tarification de l'intégration de OpsCenter avec Security Hub :
>
> - Si vous êtes connecté au compte d'administrateur Security Hub lorsque vous configurez OpsCenter et l'intégration de Security Hub, le système crée des OpsItems pour les résultats dans le compte d'administrateur et dans tous les comptes membres. Les

OpsItems sont tous créés dans le compte administrateur. En fonction de divers facteurs, cela peut entraîner une facture étonnamment élevée AWS.

Si vous êtes connecté à un compte membre lorsque vous configurez l'intégration, le système crée uniquement des OpsItems pour les résultats de ce compte individuel. Pour plus d'informations sur le compte administrateur du Security Hub, les comptes des membres et leur relation avec le flux d' EventBridge événements contenant les résultats, consultez la section Types d'intégration avec Security Hub EventBridge dans le guide de AWS Security Hub l'utilisateur.

- Pour chaque résultat qui crée un OpsItem, le prix normal de la création de l'OpsItem vous est facturé. Vous êtes également facturé si vous modifiez le OpsItem ou si le résultat correspondant est mis à jour dans Security Hub (ce qui déclenche une mise à jour du OpsItem).

Pour configurer OpsCenter afin de créer OpsItems pour les résultats de Security Hub

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez OpsCenter.
3. Sélectionnez Settings (Paramètres).
4. Dans la section Résultats de Security Hub, sélectionnez Modifier.
5. Cliquez sur le curseur pour passer de Désactivé à Activé.
6. Si vous voulez que le système crée des OpsItems pour les résultats de gravité moyenne ou faible, activez ces options.
7. Choisissez Save (Enregistrer) pour enregistrer votre configuration.

Suivez la procédure suivante si vous ne voulez plus que le système crée des OpsItems pour les résultats de Security Hub.

Pour ne plus recevoir des OpsItems pour les résultats de Security Hub

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez OpsCenter.
3. Sélectionnez Settings (Paramètres).

4. Dans la section Résultats de Security Hub, sélectionnez Modifier.
5. Cliquez sur le curseur pour faire passer l'option Activé à Désactivé. Si vous ne parvenez pas à basculer le curseur, Security Hub n'est pas activé pour votre Compte AWS.
6. Sélectionnez Enregistrer pour sauvegarder votre configuration. OpsCenter ne créera plus de OpsItems sur la base des résultats de Security Hub.

> **Important**
>
> Un administrateur délégué de Systems Manager ou le compte AWS Organizations de gestion peut activer les recherches de Security Hub OpsCenter pour plusieurs comptes et Régions AWS en créant une synchronisation des données de ressources dansExplorer. Si la source Security Hub est activée Explorer et qu'il existe une synchronisation des données des ressources qui cible le compte membre sur lequel vous avez désactivé l'intégration de Security Hub, les paramètres sélectionnés par votre administrateur sont prioritaires. OpsCentercontinue de créer OpsItems pour les conclusions du Security Hub. OpsItemsPour arrêter de créer des résultats à des fins de Security Hub dans un compte membre ciblé par une synchronisation des données de ressources, contactez votre administrateur et demandez-lui de supprimer votre compte de la synchronisation des données des ressources ou de désactiver la source du Security Hub dansExplorer. Pour plus d'informations sur la modification des paramètres dansExplorer, voirModification de sources de données Systems Manager Explorer.

Incident Manager

Incident Manager, une fonctionnalité de AWS Systems Manager, fournit une console de gestion des incidents qui vous aide à atténuer les incidents affectant vos applications AWS hébergées et à vous rétablir en cas d'incident. Un incident est une interruption ou une réduction non planifiée de la qualité des services. Une fois Incident Manager paramétré et configuré, le système crée automatiquement des OpsItems dans OpsCenter.

Lorsque le système crée un incident dans Incident Manager, il crée également un OpsItem dans OpsCenter, et affiche l'incident comme élément connexe. S'il existe déjà des OpsItem, Incident Manager ne crée pas d'OpsItem. Ce premier OpsItem est appelé l'OpsItem parent. Si l'échelle et la portée d'un incident augmentent, vous pouvez ajouter des incidents supplémentaires à un OpsItem existant. Si nécessaire, vous pouvez créer un incident manuellement pour un OpsItem. Après la

fermeture d'un incident, vous pouvez créer une analyse dans Incident Manager pour examiner et améliorer le processus de correction pour des problèmes similaires.

Par défaut, OpsCenter s'intègre à Incident Manager. Si Incident Manager n'est pas configuré, la OpsCenter page affiche un message pour configurer Incident Manager. Lorsque Incident Manager crée un OpsItem, vous pouvez gérer et corriger l'OpsItem d'OpsCenter. Pour obtenir des instructions sur la création d'un incident pour un OpsItem, consultez Création d'un incident pour un OpsItem.

Créer OpsItems

Une fois que vous avez configuré OpsCenter, une fonctionnalité de AWS Systems Manager, que vous l'avez intégré à vos Services AWS, vos Services AWS créent automatiquement des OpsItems en fonction de règles, d'événements ou d'alarmes par défaut.

Vous pouvez consulter les statuts et les niveaux de sévérité des règles Amazon EventBridge par défaut. Si nécessaire, vous pouvez créer ou modifier ces règles depuis Amazon EventBridge. Vous pouvez également consulter les alarmes depuis Amazon CloudWatch et créer ou modifier des alarmes. À l'aide de règles et d'alarmes, vous pouvez configurer les événements pour lesquels vous souhaitez générer des OpsItems automatiquement.

Lorsque le système crée un OpsItem, celui-ci est sous statut Ouvert. Vous pouvez modifier le statut pour En cours lorsque vous commencez à examiner l'OpsItem et pour Résolu une fois que vous avez corrigé l'OpsItem. Pour plus d'informations sur la façon de configurer les alarmes et les règles dans les Services AWS pour créer des OpsItems, ainsi que sur la façon de créer des Opsitems manuellement, consultez les rubriques suivantes.

Rubriques

- Configurer des règles EventBridge pour créer des OpsItems
- Configurer les alarmes CloudWatch pour créer des OpsItems
- Créer manuellement OpsItems

Configurer des règles EventBridge pour créer des OpsItems

Quand Amazon EventBridge reçoit un événement, il crée un nouveau OpsItem sur la base des règles par défaut. Vous pouvez créer une règle ou modifier une règle existante pour définir OpsCenter comme cible d'un événement EventBridge. Pour obtenir des informations sur la création d'une nouvelle règle d'événement, veuillez vous reporter à Création d'une règle pour un Service AWS dans le Guide de l'utilisateur Amazon EventBridge.

Pour configurer une règle EventBridge afin de créer des OpsItems dans OpsCenter, procédez comme suit :

1. Ouvrez la console Amazon EventBridge à l'adresse https://console.aws.amazon.com/events/.
2. Dans le volet de navigation, sélectionnez Rules.
3. Dans la page Rules (Règles), pour Event Bus (Bus d'événement), choisissez défaut.
4. Dans Règles, choisissez une règle en cochant la case à côté de son nom.
5. Sélectionnez le nom de la règle pour ouvrir sa page de détails. Dans la section Détails de la règle, vérifiez que Statut a la valeur Activé.

   > **Note**
   >
   > Si nécessaire, vous pouvez mettre à jour le statut en utilisant Modifier dans l'angle supérieur droit de la page.

6. Choisissez l'onglet Cibles.
7. Dans l'onglet Targets, choisissez Edit.
8. Pour les types de cibles, sélectionnez Service AWS.
9. Pour Select a target (Sélectionnez une cible), choisissez Systems Manager OpsItem.
10. Pour de nombreux types de cibles, EventBridge a besoin d'une autorisation pour envoyer des événements à la cible. Dans ce cas, EventBridge peut créer le rôle AWS Identity and Access Management (IAM) nécessaire à l'exécution de votre règle :

    - Pour créer un rôle IAM automatiquement, sélectionnez Create a new role for this specific resource.
    - Pour utiliser un rôle IAM que vous avez créé pour accorder à EventBridge l'autorisation de créer OpsItems dans OpsCenter, choisissez Use existing role (Utiliser un rôle existant).

11. Dans la section Réglages supplémentaires, pour Configurer l'entrée cible, choisissez Transformateur d'entrée.

    Vous pouvez utiliser l'option Transformateur d'entrée pour spécifier une chaîne de déduplication et d'autres informations importantes pour les OpsItems, telles qu'un titre et une sévérité.

12. Choisissez Configure input transformer (Configurer le transformateur d'entrée).
13. Dans la section Transformateur d'entrée cible, pour Chemin d'entrée, spécifiez les valeurs à analyser depuis l'événement déclencheur. Par exemple, pour analyser l'heure de début, l'heure de fin et d'autres détails de l'événement déclencheur de la règle, utilisez le code JSON suivant.

```
{
    "end-time": "$.detail.EndTime",
    "failure-cause": "$.detail.cause",
    "resources": "$.resources",
    "source": "$.detail.source",
    "start-time": "$.detail.StartTime"
}
```

14. Pour Template (Modèle), spécifiez les informations à envoyer à la cible. Par exemple, utilisez le code JSON suivant pour transmettre des informations à OpsCenter. Les informations sont utilisées pour créer un OpsItem.

> **Note**
>
> Si le modèle d'entrée est au format JSON, la valeur de l'objet dans le modèle ne peut pas inclure de guillemets. Par exemple, les valeurs des ressources, de la cause de l'échec, de la source, de l'heure de début et de l'heure de fin ne peuvent pas être entre guillemets.

```
{
    "title": "EBS snapshot copy failed",
    "description": "CloudWatch Event Rule SSMOpsItems-EBS-snapshot-copy-failed was
 triggered. Your EBS snapshot copy has failed. See below for more details.",
    "category": "Availability",
    "severity": "2",
    "source": "EC2",
    "resources": <resources>,
    "operationalData": {
        "/aws/dedup": {
            "type": "SearchableString",
            "value": "{\"dedupString\":\"SSMOpsItems-EBS-snapshot-copy-failed\"}"
        },
        "/aws/automations": {
            "value": "[ { \"automationType\": \"AWS:SSM:Automation\",
 \"automationId\": \"AWS-CopySnapshot\" } ]"
        },
        "failure-cause": {
            "value": <failure-cause>
        },
```

```
        "source": {
            "value": <source>
        },
        "start-time": {
            "value": <start-time>
        },
        "end-time": {
            "value": <end-time>
        }
    }
}
```

Pour de plus amples informations sur ces champs, consultez Transformation de l'entrée cible dans le Guide de l'utilisateur Amazon EventBridge.

15. Choisissez Confirm (Confirmer).
16. Choisissez Next (Suivant).
17. Choisissez Next (Suivant).
18. Choisissez Mettre à jour la règle.

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 Gestion des OpsItems.

Configurer les alarmes CloudWatch pour créer des OpsItems

Lors de la configuration intégrée d'OpsCenter, une fonctionnalité AWS Systems Manager, vous permettez à Amazon CloudWatch de créer automatiquement des OpsItems sur la base des alarmes courantes. Vous pouvez créer une alarme ou modifier une alarme existante pour créer des OpsItems dans OpsCenter.

CloudWatch crée un nouveau rôle lié au service dans AWS Identity and Access Management (IAM) lorsque vous configurez une alarme pour créer des OpsItems. Le nouveau rôle est nommé `AWSServiceRoleForCloudWatchAlarms_ActionSSM`. Pour de plus amples informations sur les rôles liés au service CloudWatch, consultez Utilisation de rôles liés au service pour CloudWatch dans le Guide de l'utilisateur Amazon CloudWatch..

Lorsqu'une alarme CloudWatch génère un OpsItem, l'OpsItem affiche alarme CloudWatch - « ***alarm_name*** » est à l'état ALARM.

Pour afficher des détails sur un OpsItem particulier, sélectionnez l'OpsItem, puis sélectionnez l'onglet Détails des ressources connexes. Vous pouvez modifier manuellement des OpsItems pour modifier des détails tels que la sévérité ou la catégorie. Toutefois, lorsque vous modifiez la sévérité ou la catégorie d'une alarme, Systems Manager ne peut pas mettre à jour la sévérité ou la catégorie d'OpsItems déjà créés à partir de l'alarme. Si une alarme a créé un OpsItem et que vous avez spécifié une chaîne de déduplication, l'alarme ne créera pas d'OpsItems supplémentaires même si vous modifiez l'alarme dans CloudWatch. Si l'OpsItem est résolu dans OpsCenter, CloudWatch créera un nouvel OpsItem.

Pour plus d'informations sur les métriques CloudWatch, consultez les rubriques suivantes.

Rubriques

- Configuration d'une alarme CloudWatch pour créer des OpsItems (console)
- Configuration d'une alarme CloudWatch existante pour créer des OpsItems (par programmation)

Configuration d'une alarme CloudWatch pour créer des OpsItems (console)

Vous pouvez créer une alarme manuellement ou mettre à jour une alarme existante pour créer des OpsItems à partir d'Amazon CloudWatch.

Pour créer une alarme CloudWatch et configurer Systems Manager comme la cible de cette alarme.

1. Suivez les étapes 1 à 9 telles qu'indiquées dans Création d'une alarme CloudWatch basée sur un seuil statique dans le Guide de l'utilisateur Amazon CloudWatch.
2. Dans la section Action Systems Manager, sélectionnez Ajouter une action Systems Manager OpsCenter.
3. Choisissez OpsItems.
4. Pour Sévérité, sélectionnez une valeur de 1 à 4.
5. (Facultatif) Pour Catégorie, sélectionnez une catégorie pour l'OpsItem.
6. Suivez les étapes 11 à 13 telles qu'indiquées dans Création d'une alarme CloudWatch basée sur un seuil statique dans le Guide de l'utilisateur Amazon CloudWatch.
7. Sélectionnez Next (Suivant) et suivez l'assistant.

Procédez comme suit pour modifier une alarme existante et configurer Systems Manager comme la cible de cette alarme.

1. Ouvrez la console CloudWatch à l'adresse https://console.aws.amazon.com/cloudwatch/.
2. Dans le panneau de navigation, cliquez sur Alarms (alertes).
3. Sélectionnez l'alarme, puis Actions, Edit (Modifier).
4. (Facultatif) Modifiez les paramètres dans les sections Metrics (Métriques) et Conditions, puis sélectionnez Next (Suivant).
5. Dans la section Systems Manager, sélectionnez Add Systems Manager OpsCenter action (Ajouter une action Systems Manager OpsCenter).
6. Pour Severity (Sévérité), sélectionnez un nombre.

> **Note**
>
> La sévérité est une valeur définie par l'utilisateur. Vous ou votre organisation déterminez la signification de chaque valeur de sévérité et les accords de niveau de service associés à chaque sévérité.

7. (Facultatif) Pour Category (Catégorie), sélectionnez une option.
8. Sélectionnez Next (Suivant) et suivez l'assistant.

Configuration d'une alarme CloudWatch existante pour créer des OpsItems (par programmation)

Vous pouvez configurer des alarmes Amazon CloudWatch pour créer par programmation des OpsItems en utilisant l'AWS Command Line Interface (AWS CLI), des modèles AWS CloudFormation ou des extraits de code Java.

Rubriques

- Avant de commencer
- Configuration des alarmes CloudWatch pour créer des OpsItems (AWS CLI)
- Configuration des alarmes CloudWatch pour créer ou mettre à jour des OpsItems (CloudFormation)
- Configuration d'alarmes CloudWatch pour créer ou mettre à jour des OpsItems (Java)

Avant de commencer

Si vous modifiez une alarme existante par programmation ou que vous créez une nouvelle alarme créant des OpsItems, vous devez spécifier un Amazon Resource Name (ARN). Cet ARN identifie Systems Manager OpsCenter comme la cible des OpsItems créés à partir de l'alarme. Vous pouvez personnaliser l'ARN de sorte que des OpsItems créés à partir de l'alarme incluent des informations spécifiques telles que la sévérité ou la catégorie. Chaque ARN inclut les informations décrites dans le tableau suivant.

Paramètre	Détails
`Region` (obligatoire)	La Région AWS où l'alarme existe. Par exemple : `us-west-2` . Pour obtenir des informations sur les Régions AWS dans lesquelles utiliser OpsCenter, consultez AWS Systems ManagerPoints de terminaison et quotas.
`account_ID` (obligatoire)	Le même cas ID Compte AWS que celui utilisé pour créer l'alarme. Par exemple : `123456789012` . L'ID de compte doit être suivi du signe deux-points (`:`) et du paramètre `opsitem`, comme le montrent les exemples suivants.
`severity` (obligatoire)	Un niveau de sévérité défini par l'utilisateur pour les OpsItems créés à partir de l'alarme. Valeurs valides : `1`, `2`, `3`, `4`
`Category` (facultatif)	Une catégorie pour des OpsItems créés à partir de l'alarme. Valeurs valides : `Availability` , `Cost`, `Performance` , `Recovery` et `Security`.

Créez l'ARN en utilisant la syntaxe suivante. Cet ARN n'inclut pas le paramètre `Category` facultatif.

```
arn:aws:ssm:Region:account_ID:opsitem:severity
```

Voici un exemple.

```
arn:aws:ssm:us-west-2:123456789012:opsitem:3
```

Pour créer un ARN utilisant le paramètre `Category` facultatif, respectez la syntaxe suivante.

```
arn:aws:ssm:Region:account_ID:opsitem:severity#CATEGORY=category_name
```

Voici un exemple.

```
arn:aws:ssm:us-west-2:123456789012:opsitem:3#CATEGORY=Security
```

Configuration des alarmes CloudWatch pour créer des OpsItems (AWS CLI)

Cette commande exige que vous spécifiiez un ARN pour le paramètre `alarm-actions`. Pour de plus amples informations sur la création de l'ARN, consultez Avant de commencer.

Configurer CloudWatch pour créer des OpsItems (AWS CLI)

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS Command Line Interface (AWS CLI).

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l'AWS CLI.
2. Exécutez la commande suivante pour collecter des informations sur l'alarme à configurer.

   ```
   aws cloudwatch describe-alarms --alarm-names "alarm name"
   ```
3. Exécutez la commande suivante pour mettre à jour une alarme. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   ```
   aws cloudwatch put-metric-alarm --alarm-name name \
   --alarm-description "description" \
   --metric-name name --namespace namespace \
   --statistic statistic --period value --threshold value \
   --comparison-operator value \
   --dimensions "dimensions" --evaluation-periods value \
       --alarm-actions
    arn:aws:ssm:Region:account_ID:opsitem:severity#CATEGORY=category_name \
   --unit unit
   ```

   Voici un exemple :

Linux & macOS

```
aws cloudwatch put-metric-alarm --alarm-name cpu-mon \
--alarm-description "Alarm when CPU exceeds 70 percent" \
--metric-name CPUUtilization --namespace AWS/EC2 \
--statistic Average --period 300 --threshold 70 \
--comparison-operator GreaterThanThreshold \
--dimensions "Name=InstanceId,Value=i-12345678" --evaluation-periods 2 \
--alarm-actions arn:aws:ssm:us-east-1:123456789012:opsitem:3#CATEGORY=Security \
--unit Percent
```

Windows

```
aws cloudwatch put-metric-alarm --alarm-name cpu-mon ^
--alarm-description "Alarm when CPU exceeds 70 percent" ^
--metric-name CPUUtilization --namespace AWS/EC2 ^
--statistic Average --period 300 --threshold 70 ^
--comparison-operator GreaterThanThreshold ^
--dimensions "Name=InstanceId,Value=i-12345678" --evaluation-periods 2 ^
--alarm-actions arn:aws:ssm:us-east-1:123456789012:opsitem:3#CATEGORY=Security ^
--unit Percent
```

Configuration des alarmes CloudWatch pour créer ou mettre à jour des OpsItems (CloudFormation)

Cette section inclut des modèles AWS CloudFormation que vous pouvez utiliser pour configurer des alarmes CloudWatch afin de créer ou de mettre à jour automatiquement des OpsItems. Chaque modèle de code exige que vous spécifiiez un Amazon Resource Name (ARN) pour le paramètre `AlarmActions`. Pour de plus amples informations sur la création de l'ARN, consultez Avant de commencer.

Alarme de métrique – Utilisez le modèle CloudFormation suivant pour créer ou mettre à jour une alarme de métrique CloudWatch. L'alarme spécifiée dans ce modèle contrôle les vérifications du statut d'instances Amazon Elastic Compute Cloud (Amazon EC2). Si l'alarme entre dans l'état `ALARM`, un OpsItem est créé dans OpsCenter.

```
{
  "AWSTemplateFormatVersion": "2010-09-09",
```

```
      "Parameters" : {
        "RecoveryInstance" : {
          "Description" : "The EC2 instance ID to associate this alarm with.",
          "Type" : "AWS::EC2::Instance::Id"
        }
      },
      "Resources": {
        "RecoveryTestAlarm": {
          "Type": "AWS::CloudWatch::Alarm",
          "Properties": {
            "AlarmDescription": "Run a recovery action when instance status check fails
 for 15 consecutive minutes.",
            "Namespace": "AWS/EC2" ,
            "MetricName": "StatusCheckFailed_System",
            "Statistic": "Minimum",
            "Period": "60",
            "EvaluationPeriods": "15",
            "ComparisonOperator": "GreaterThanThreshold",
            "Threshold": "0",
            "AlarmActions": [ {"Fn::Join" : ["",
 ["arn:arn:aws:ssm:Region:account_ID:opsitem:severity#CATEGORY=category_name",
 { "Ref" : "AWS::Partition" }, ":ssm:", { "Ref" : "AWS::Region" }, { "Ref" : "AWS::
 AccountId" }, ":opsitem:3" ]]} ],
            "Dimensions": [{"Name": "InstanceId","Value": {"Ref": "RecoveryInstance"}}]
          }
        }
      }
    }
```

Alarme composite – Utilisez le modèle CloudFormation suivant pour créer ou mettre à jour une alarme composite. Une alarme composite est constituée de plusieurs alarmes de métrique. Si l'alarme entre dans l'état ALARM, un OpsItem est créé dans OpsCenter.

```
"Resources":{
       "HighResourceUsage":{
          "Type":"AWS::CloudWatch::CompositeAlarm",
          "Properties":{
             "AlarmName":"HighResourceUsage",
             "AlarmRule":"(ALARM(HighCPUUsage) OR ALARM(HighMemoryUsage)) AND NOT
 ALARM(DeploymentInProgress)",

 "AlarmActions":"arn:aws:ssm:Region:account_ID:opsitem:severity#CATEGORY=category_name",
```

```
            "AlarmDescription":"Indicates that the system resource usage is high while
no known deployment is in progress"
         },
         "DependsOn":[
            "DeploymentInProgress",
            "HighCPUUsage",
            "HighMemoryUsage"
         ]
      },
      "DeploymentInProgress":{
         "Type":"AWS::CloudWatch::CompositeAlarm",
         "Properties":{
            "AlarmName":"DeploymentInProgress",
            "AlarmRule":"FALSE",
            "AlarmDescription":"Manually updated to TRUE/FALSE to disable other
alarms"
         }
      },
      "HighCPUUsage":{
         "Type":"AWS::CloudWatch::Alarm",
         "Properties":{
            "AlarmDescription":"CPUusageishigh",
            "AlarmName":"HighCPUUsage",
            "ComparisonOperator":"GreaterThanThreshold",
            "EvaluationPeriods":1,
            "MetricName":"CPUUsage",
            "Namespace":"CustomNamespace",
            "Period":60,
            "Statistic":"Average",
            "Threshold":70,
            "TreatMissingData":"notBreaching"
         }
      },
      "HighMemoryUsage":{
         "Type":"AWS::CloudWatch::Alarm",
         "Properties":{
            "AlarmDescription":"Memoryusageishigh",
            "AlarmName":"HighMemoryUsage",
            "ComparisonOperator":"GreaterThanThreshold",
            "EvaluationPeriods":1,
            "MetricName":"MemoryUsage",
            "Namespace":"CustomNamespace",
            "Period":60,
            "Statistic":"Average",
```

```
            "Threshold":65,
            "TreatMissingData":"breaching"
          }
       }
    }
```

Configuration d'alarmes CloudWatch pour créer ou mettre à jour des OpsItems (Java)

Cette section inclut des extraits de code Java que vous pouvez utiliser pour configurer des alarmes CloudWatch afin de créer ou de mettre à jour automatiquement des OpsItems. Chaque extrait exige que vous spécifiiez un ARN pour le paramètre `validSsmActionStr`. Pour de plus amples informations sur la création de l'ARN, consultez Avant de commencer.

Une alarme spécifique – Utilisez l'extrait de code Java suivant pour créer ou mettre à jour une alarme CloudWatch. L'alarme spécifiée dans ce modèle contrôle les vérifications du statut d'instances Amazon EC2. Si l'alarme entre dans l'état ALARM, un OpsItem est créé dans OpsCenter.

```
import com.amazonaws.services.cloudwatch.AmazonCloudWatch;
    import com.amazonaws.services.cloudwatch.AmazonCloudWatchClientBuilder;
    import com.amazonaws.services.cloudwatch.model.ComparisonOperator;
    import com.amazonaws.services.cloudwatch.model.Dimension;
    import com.amazonaws.services.cloudwatch.model.PutMetricAlarmRequest;
    import com.amazonaws.services.cloudwatch.model.PutMetricAlarmResult;
    import com.amazonaws.services.cloudwatch.model.StandardUnit;
    import com.amazonaws.services.cloudwatch.model.Statistic;

    private void putMetricAlarmWithSsmAction() {
        final AmazonCloudWatch cw =
                AmazonCloudWatchClientBuilder.defaultClient();

        Dimension dimension = new Dimension()
                .withName("InstanceId")
                .withValue(instanceId);

        String validSsmActionStr =
 "arn:aws:ssm:Region:account_ID:opsitem:severity#CATEGORY=category_name";

        PutMetricAlarmRequest request = new PutMetricAlarmRequest()
                .withAlarmName(alarmName)
                .withComparisonOperator(
                        ComparisonOperator.GreaterThanThreshold)
                .withEvaluationPeriods(1)
                .withMetricName("CPUUtilization")
```

```
                .withNamespace("AWS/EC2")
                .withPeriod(60)
                .withStatistic(Statistic.Average)
                .withThreshold(70.0)
                .withActionsEnabled(false)
                .withAlarmDescription(
                        "Alarm when server CPU utilization exceeds 70%")
                .withUnit(StandardUnit.Seconds)
                .withDimensions(dimension)
                .withAlarmActions(validSsmActionStr);

        PutMetricAlarmResult response = cw.putMetricAlarm(request);
    }
```

Mettre à jour toutes les alarmes – Utilisez l'extrait de code Java suivant pour mettre à jour toutes les alarmes CloudWatch dans votre Compte AWS afin de créer des OpsItems lorsqu'une alarme entre dans l'état ALARM.

```
import com.amazonaws.services.cloudwatch.AmazonCloudWatch;
    import com.amazonaws.services.cloudwatch.AmazonCloudWatchClientBuilder;
    import com.amazonaws.services.cloudwatch.model.DescribeAlarmsRequest;
    import com.amazonaws.services.cloudwatch.model.DescribeAlarmsResult;
    import com.amazonaws.services.cloudwatch.model.MetricAlarm;

    private void listMetricAlarmsAndAddSsmAction() {
        final AmazonCloudWatch cw = AmazonCloudWatchClientBuilder.defaultClient();

        boolean done = false;
        DescribeAlarmsRequest request = new DescribeAlarmsRequest();

        String validSsmActionStr =
 "arn:aws:ssm:Region:account_ID:opsitem:severity#CATEGORY=category_name";

        while(!done) {

            DescribeAlarmsResult response = cw.describeAlarms(request);

            for(MetricAlarm alarm : response.getMetricAlarms()) {
                // assuming there are no alarm actions added for the metric alarm
                alarm.setAlarmActions(ImmutableList.of(validSsmActionStr));
            }

            request.setNextToken(response.getNextToken());
```

```
            if(response.getNextToken() == null) {
                done = true;
            }
        }
    }
```

Créer manuellement OpsItems

Lorsque vous détectez un problème opérationnel, vous pouvez créer manuellement un OpsItem depuis OpsCenter (fonctionnalité développée par AWS Systems Manager) afin de gérer et de résoudre le problème.

Si vous configurez OpsCenter pour une administration multicomptes, un compte administrateur délégué Systems Manager ou un compte de gestion AWS Organizations peut créer des OpsItems pour les comptes membre. Pour de plus amples informations, veuillez consulter (Facultatif) Configuration d'OpsCenter pour gérer de manière centralisée les OpsItems de l'ensemble des comptes.

Vous pouvez créer des OpsItems à l'aide de la console AWS Systems Manager, de l'AWS Command Line Interface (AWS CLI) ou d'AWS Tools for Windows PowerShell.

Rubriques

- Création manuelle d'OpsItems (console)
- Création manuelle d'OpsItems (AWS CLI)
- Création manuelle d'OpsItems (PowerShell)

Création manuelle d'OpsItems (console)

Vous pouvez créer des OpsItems manuellement à l'aide de la console AWS Systems Manager. Lorsque vous créez un OpsItem, il s'affiche dans votre compte OpsCenter. Si vous configurez OpsCenter pour l'administration multicomptes, OpsCenter fournit à l'administrateur délégué ou au compte de gestion la possibilité de créer des OpsItems pour les comptes membre sélectionnés. Pour de plus amples informations, veuillez consulter (Facultatif) Configuration d'OpsCenter pour gérer de manière centralisée les OpsItems de l'ensemble des comptes.

Créer un OpsItem à l'aide de 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, sélectionnez OpsCenter.
3. Sélectionnez Créer un OpsItem. Si vous ne voyez pas ce bouton, cliquez sur l'onglet OpsItems, puis sélectionnez Create OpsItem (Créer un OpsItem).
4. (Facultatif) Choisissez Autre compte, puis choisissez le compte pour lequel vous souhaitez créer l'OpsItem.

   > **Note**
   >
   > Cette étape est obligatoire si vous créez des OpsItems pour un compte membre.

5. Dans Titre, saisissez un nom descriptif pour vous aider à comprendre l'objectif de l'OpsItem.
6. 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.

7. (Facultatif) Pour Priority (Priorité), sélectionnez le niveau de priorité.
8. (Facultatif) Pour Severity (Sévérité), sélectionnez le niveau de sévérité.
9. (Facultatif) Pour Category (Catégorie), sélectionnez une catégorie.
10. Pour Description, saisissez des informations à propos de cet OpsItem y compris (le cas échéant) les étapes de reproduction du problème.

    > **Note**
    >
    > La console prend en charge la plupart des formats markdown dans le champ de description OpsItem. Pour plus d'informations, consultez Utilisation de Markdown dans la console du Guide de mise en route de la AWS Management Console.

11. Pour Chaîne de déduplication, saisissez les mots que le système peut utiliser pour rechercher les OpsItems en double. Pour plus d'informations sur les chaînes de déduplication, consultez Gestion des OpsItems en double.

12. (Facultatif) Pour Notifications, spécifiez l'Amazon Resource Name (ARN) de la rubrique Amazon SNS dans laquelle vous souhaitez envoyer des notifications lorsque cet OpsItem est mis à jour. Vous devez spécifier un ARN Amazon SNS qui se trouve dans la même Région AWS que l'OpsItem.
13. (Facultatif) Sous Ressources connexes, choisissez Ajouter pour spécifier l'ID ou l'ARN de la ressource affectée et de toute ressource connexe.
14. Sélectionnez Créer un OpsItem.

En cas de succès, la page affiche l'OpsItem. Lorsqu'un compte d'administrateur ou de gestion délégué crée un OpsItem pour des comptes membre sélectionnés, les nouveaux OpsItems sont affichés dans l'OpsCenter de l'administrateur et des membres. Pour plus d'informations sur la configuration des options dans un OpsItem, consultez Gestion des OpsItems.

Création manuelle d'OpsItems (AWS CLI)

La procédure suivante décrit comment créer un clone OpsItem à l'aide de l'AWS Command Line Interface (AWS CLI).

Pour créer un OpsItem à l'aide du AWS CLI

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS Command Line Interface (AWS CLI).

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l'AWS CLI.
2. Ouvrez la AWS CLI et exécutez la commande suivante pour créer un OpsItem. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   ```
   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"
   ```

   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.

```
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 plus d'informations sur la saisie de paramètres au format JSON sur la ligne de commande sous différents systèmes d'exploitation locaux, consultez Utilisation de guillemets avec des chaînes dans la AWS CLI dans le Guide de l'utilisateur AWS Command Line Interface.

Le système retourne des informations telles que les suivantes.

```
{
    "OpsItemId": "oi-1a2b3c4d5e6f"
}
```

3. Maintenant, exécutez la commande suivante pour afficher les détails relatifs à l'OpsItem que vous avez créé.

```
aws ssm get-ops-item --ops-item-id ID
```

Le système retourne des informations telles que les suivantes.

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

4. 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.

5. Exécutez à nouveau la commande suivante pour vérifier que le statut est passé à `InProgress`.

```
aws ssm get-ops-item --ops-item-id ID
```

Exemples de création d'un OpsItem

Les exemples de code suivants vous montrent comment créer un OpsItem à l'aide du portail de gestion Linux, macOS ou Windows.

Portail de gestion Linux ou macOS

La commande suivante crée un OpsItem quand un disque d'instance Amazon Elastic Compute Cloud (Amazon EC2) est plein.

```
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

La commande suivante crée un OpsItem lorsqu'une instance Amazon Relational Database Service (Amazon RDS) ne répond pas.

```
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 connexe de 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 runbook `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\"}}
```

Création manuelle d'OpsItems (PowerShell)

La procédure suivante décrit comment créer un OpsItem à l'aide de AWS Tools for Windows PowerShell.

Créer un OpsItem à l'aide de AWS Tools for Windows PowerShell

1. Ouvrez les AWS Tools for Windows PowerShell et exécutez la commande suivante pour spécifier vos informations d'identification.

   ```
   Set-AWSCredentials –AccessKey key-name –SecretKey key-name
   ```

2. Exécutez la commande suivante pour définir laRégion AWS de votre session PowerShell.

   ```
   Set-DefaultAWSRegion -Region Region
   ```

3. Exécutez la commande suivante pour créer un nouveau OpsItem. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations. Cette commande spécifie un runbook Systems Manager Automation pour corriger cet OpsItem.

   ```
   $opsItem = New-Object Amazon.SimpleSystemsManagement.Model.OpsItemDataValue
   $opsItem.Type = [Amazon.SimpleSystemsManagement.OpsItemDataType]::SearchableString
   $opsItem.Value = '[{\"automationId\":\"runbook_name\",\"automationType\":
   \"AWS::SSM::Automation\"}]'
   $newHash = @{" /aws/
   automations"=[Amazon.SimpleSystemsManagement.Model.OpsItemDataValue]$opsItem}

   New-SSMOpsItem `
       -Title "title" `
       -Description "description" `
       -Priority priority_number `
       -Source AWS_service `
       -OperationalData $newHash
   ```

   En cas de réussite, la commande génère l'ID du nouveau OpsItem.

L'exemple suivant spécifie l'Amazon Resource Name (ARN) d'une instance Amazon Elastic Compute Cloud (Amazon EC2).

```
$opsItem = New-Object Amazon.SimpleSystemsManagement.Model.OpsItemDataValue
$opsItem.Type = [Amazon.SimpleSystemsManagement.OpsItemDataType]::SearchableString
$opsItem.Value = '[{\"arn\":\"arn:aws:ec2:us-east-1:123456789012:instance/
i-1234567890abcdef0\"}]'
$newHash = @{" /aws/
resources"=[Amazon.SimpleSystemsManagement.Model.OpsItemDataValue]$opsItem}
New-SSMOpsItem -Title "EC2 instance disk full still" -Description "Log clean up may
 have failed which caused the disk to be full" -Priority 2 -Source ec2 -OperationalData
 $newHash
```

Gestion des OpsItems

OpsCenter est une fonctionnalité de AWS Systems Manager qui assure le suivi des OpsItems de leur création jusqu'à leur résolution. Si vous configurez OpsCenter pour une administration entre comptes, un administrateur délégué ou un compte de gestion peut gérer les OpsItems à partir de son compte. Pour de plus amples informations, veuillez consulter (Facultatif) Configuration d'OpsCenter pour gérer de manière centralisée les OpsItems de l'ensemble des comptes.

Vous pouvez consulter et gérer les OpsItems à l'aide des pages suivantes de la console Systems Manager :

- Résumé – Affiche le nombre d'OpsItems ouverts et en cours, le nombre d'OpsItems par source et par âge, ainsi que des informations opérationnelles. Vous pouvez filtrer les OpsItems par source et par statut d'OpsItems.

- OpsItems – Affiche une liste des OpsItems avec plusieurs champs d'information tels que le titre, l'identifiant, la priorité, la description, la source de l'OpsItem, ainsi que la date et l'heure de la dernière mise à jour. Cette page vous permet de créer manuellement des OpsItems, de configurer des sources, de modifier le statut d'un OpsItem et de filtrer les OpsItems en fonction des nouveaux incidents. Vous pouvez choisir un OpsItem pour afficher sa page Détails des OpsItems.

- Détails des OpsItem – Affiche les informations détaillées et les outils qui peuvent vous aider à gérer un OpsItem. La page Détails des OpsItems comporte les onglets suivants :
  - Présentation – Affiche les ressources connexes, les runbooks exécutés au cours des 30 derniers jours et la liste des runbooks disponibles que vous pouvez exécuter. Vous pouvez également

afficher des OpsItems similaires, ajouter des données opérationnelles et ajouter des OpsItems connexes.

- Détails des ressources connexes – Affiche des informations sur la ressource provenant 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).

Pour plus d'informations sur la façon de gérer les OpsItems, consultez les rubriques suivantes.

Rubriques

Affichage des détails d'un OpsItem.

Pour afficher complètement un OpsItem, utilisez la page de Détails de l'OpsItem sur la console OpsCenter. La page Présentation affiche les informations suivantes :

- Détails des OpsItems – Affiche des informations générales sur l'OpsItem sélectionné.
- Ressources connexes – Une ressource connexe est une ressource affectée ou la ressource qui a déclenché l'événement ayant créé l'OpsItem.
- Exécutions automatisées au cours des 30 derniers jours – Liste des runbooks exécutés au cours des 30 derniers jours.
- Runbooks – Vous pouvez choisir un runbook dans la liste.

- OpsItems similaires – Il s'agit d'une liste générée par le système avec les OpsItems qui pourraient être connexes 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.
- Données opérationnelles – Les données opérationnelles sont des données personnalisées qui contiennent des références sur l'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.
- OpsItems connexes – Vous pouvez spécifier les identifiants des OpsItems qui sont d'une manière ou d'une autre connexes à l'OpsItem actuel.
- Détails des ressources connexes – Affiche les fournisseurs de données comme les métriques et les alarmes Amazon CloudWatch, les journaux AWS CloudTrail ou les détails issus de AWS Config.

Pour afficher les détails d'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, sélectionnez OpsCenter.
3. Choisissez un OpsItem pour en afficher les détails.

Modification d'un OpsItem

La section Détails sur l'OpsItem affiche des informations concernant un OpsItem comme sa description, son titre, sa source, son ID d'OpsItem et son état.
Vous pouvez modifier un seul OpsItem, ou sélectionner plusieurs OpsItems pour modifier les champs suivants : État, Priorité, Sévérité et Catégorie.

Lorsqu'Amazon EventBridge crée unOpsItem, il remplit les champs Titre, Source et Description. Vous pouvez modifier les champs Titre et Description, mais vous ne pouvez pas modifier le champ Source.

> **Note**
>
> La console prend en charge la plupart des mises en forme du Markdown dans le champ de OpsItem description. Pour plus d'informations, consultez la section Utilisation de Markdown dans la console dans le guide de AWS Management Console démarrage.

Vous pouvez généralement modifier les données configurables suivantes pour un OpsItem :

- Titre – nom de l'OpsItem. La source crée le titre de l'OpsItem.
- Description : informations sur cet OpsItem, y compris (le cas échéant) les étapes pour reproduire le problème.
- Statut – Le statut d'un OpsItem peut être Ouvert, En cours ou Résolu.
- Priorité – Le niveau de priorité d'un OpsItem peut aller de 1 à 5. Nous recommandons que votre organisation détermine la signification de chaque niveau de priorité ainsi que l'accord de niveau de service associé à chacun.
- Sévérité – Le niveau de sévérité d'un OpsItem peut aller de 1 à 4, 1 correspondant au niveau critique, 2 au niveau élevé, 3 au niveau moyen et 4 au niveau faible.
- Catégorie – Un OpsItem peut être catégorisé en fonction de sa disponibilité, de son coût, de ses performances, de ses options de récupération ou de sa sécurité.
- Notifications – Lorsque vous modifiez un OpsItem, vous pouvez spécifier l'Amazon Resource Name (ARN) d'une rubrique Amazon Simple Notification Service 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. Pour de plus amples informations, consultez dans le Guide du développeur Amazon Simple Notification Service.

> **Important**
>
> La rubrique Amazon SNS doit exister au même Région AWS titre que le. OpsItem Si la rubrique et l'OpsItem se trouvent dans des régions différentes, le système renvoie une erreur.

OpsCenterpossède une intégration bidirectionnelle avec AWS Security Hub. Lorsque vous mettez à jour l'état et le niveau de sévérité d'un OpsItem selon un résultat de sécurité, ces modifications sont automatiquement envoyées à Security Hub afin que les informations affichées soient toujours correctes et à jour.

Lorsqu'un OpsItem est créé à partir d'une découverte du Security Hub, les métadonnées du Security Hub sont automatiquement ajoutées au champ de données opérationnelles duOpsItem. Si ces métadonnées sont supprimées, les mises à jour bidirectionnelles ne fonctionnent plus.

Pour modifier les détails de OpsItem

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.

2. Dans le panneau de navigation, sélectionnez OpsCenter.
3. Sélectionnez un ID d'OpsItem pour ouvrir la page des détails, ou sélectionnez plusieurs OpsItems. Si vous sélectionnez plusieurs OpsItems, vous pouvez uniquement modifier l'état, la priorité, la sévérité ou la catégorie. Si vous modifiez plusieurs OpsItems, OpsCenter met à jour et enregistre vos modifications dès que vous sélectionnez le nouvel état, la nouvelle priorité, la nouvelle sévérité ou la nouvelle catégorie.
4. Dans la section des OpsItem détails, choisissez Modifier.
5. Modifiez les détails de la OpsItem conformément aux exigences et recommandations spécifiées par votre organisation.
6. Lorsque vous avez terminé, sélectionnez Enregistrer.

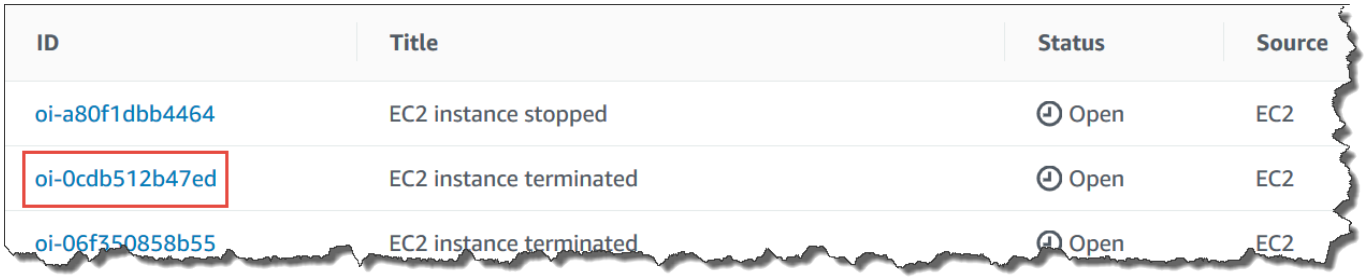
Ajout de ressources connexes à un OpsItem

Chaque OpsItem comprend une section Ressources connexes qui répertorie l'Amazon Resource Name (ARN) de la ressource connexe. Une ressource connexe est une ressource AWS affectée qui requiert un examen.

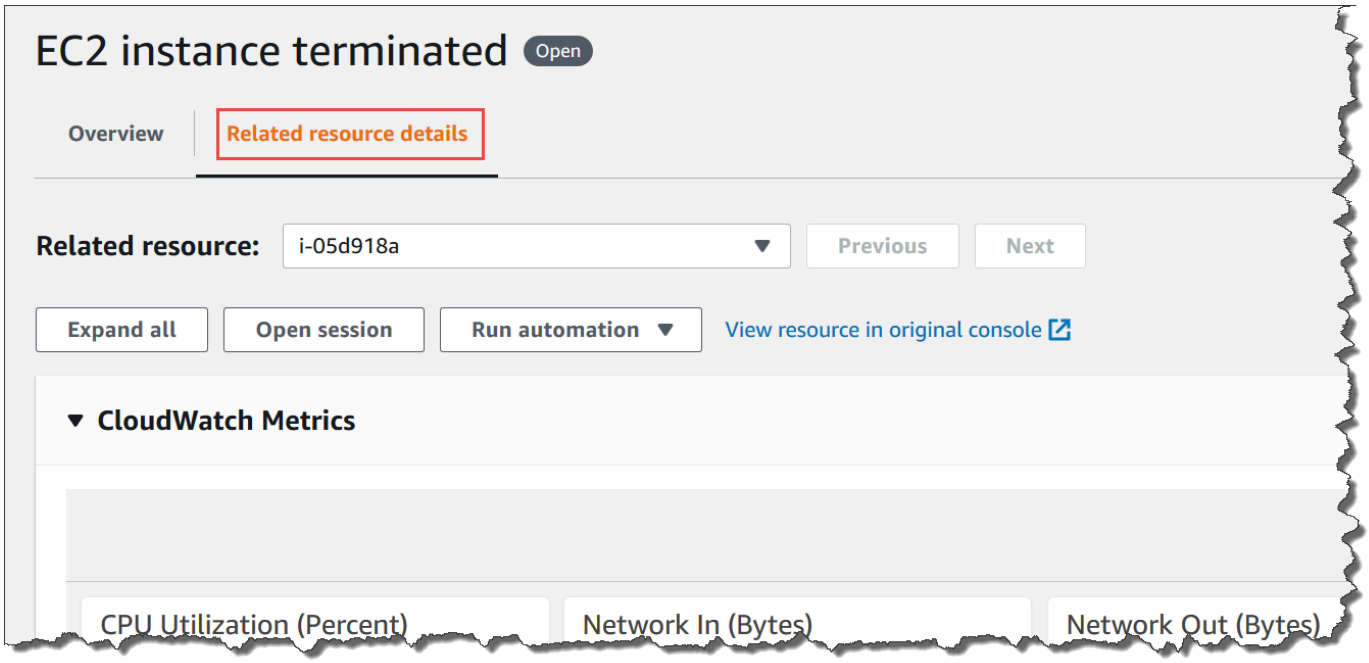
Si c'est Amazon EventBridge qui a créé l'OpsItem, le système remplit automatiquement les détails de cet OpsItem avec l'ARN de la ressource. Vous pouvez spécifier manuellement les ARN des ressources connexes. Pour certains types d'ARN, OpsCenter crée automatiquement un lien profond qui renvoie aux détails sur la ressource directement dans la console OpsCenter. Par exemple, si vous spécifiez l'ARN d'une instance Amazon Elastic Compute Cloud (Amazon EC2) en tant que ressource connexe, OpsCenter affiche les détails de cette instance EC2. Cela vous permet d'afficher des informations détaillées sur vos ressources AWS qui peuvent être affectés sans avoir à quitter OpsCenter.

Pour afficher et ajouter des ressources connexes à un OpsItem, 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, sélectionnez OpsCenter.
3. Cliquez sur l'onglet OpsItems.
4. Sélectionnez un ID OpsItem.



5. Pour afficher des informations sur les ressources impactées, sélectionnez 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, sélectionnez Ajouter.
8. Pour Type de ressource, sélectionnez une ressource dans la liste.
9. Pour ID de ressource, saisissez l'ID ou l'Amazon Resource Name (ARN). Le type d'information que vous sélectionnez dépend de la ressource que vous avez choisie à l'étape précédente.

> **Note**
>
> Vous pouvez ajouter manuellement les ARN des ressources connexes supplémentaires. Chaque OpsItem peut répertorier un maximum de 100 ARN de ressources connexes.

Le tableau suivant répertorie les types de ressources qui créent automatiquement des liens profonds vers des ressources connexes.

Types de ressources pris 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`
Distribution Amazon CloudFront	`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`

Nom de la ressource	Format ARN
AWS CodePipeline	`arn:aws:codepipeline:region:account-id:resource-specifier`
Information Amazon DevOps Guru	`arn:aws:devops-guru:region:account-id:insight/proactive or reactive/resource-id`
Table Amazon DynamoDB	`arn:aws:dynamodb:region:account-id:table/tablename`
Passerelle client Amazon Elastic Compute Cloud (Amazon EC2)	`arn:aws:ec2:region:account-id:customer-gateway/cgw-id`
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 au réseau (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`

Nom de la ressource	Format ARN
Table de routage Amazon EC2	`arn:aws:ec2: region:account-id :route-table/ route-table-id`
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 (Classic Load Balancer)	`arn:aws:elasticloadbalancing: region:account-id :loadbalancer/ name`

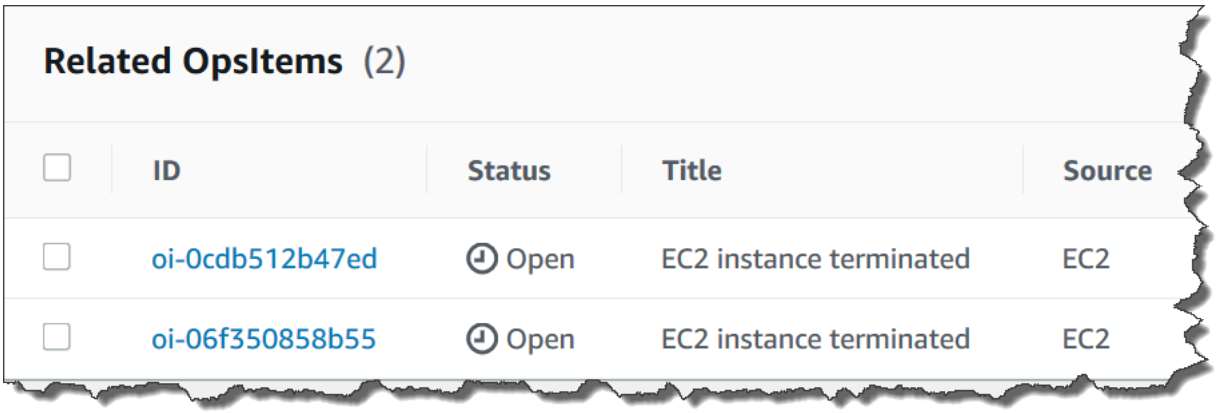
Nom de la ressource	Format ARN
Elastic Load Balancing (Application Load Balancer)	`arn:aws:elasticloadbalancing: region:account-id :loadbalancer/app/ load-balancer-name /load-balancer-id`
Elastic Load Balancing (Network Load Balancer)	`arn:aws:elasticloadbalancing: region:account-id :loadbalancer/net/ load-balancer-name /load-balancer-id`
Groupe AWS Identity and Access Management (IAM)	`arn:aws:iam:: account-id :group/group-name`
Politique IAM	`arn:aws:iam:: account-id :policy/policy-name`
Rôle IAM	`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`
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`

Nom de la ressource	Format ARN
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`

Nom de la ressource	Format ARN
Enregistrement AWS Config de l'inventaire AWS Systems Manager des nœuds gérés	`arn:aws:ssm:region:account-id:managed-instance-inventory/node_id`
Association Systems Manager State Manager	`arn:aws:ssm:region:account-id:association/association_ID`

Ajout d'OpsItems connexes à un OpsItem

La fonction OpsItems connexes disponible sur la page Détails d'OpsItems vous permet d'examiner les problèmes opérationnels et d'obtenir du contexte pour ces problèmes. Les OpsItems peuvent être connectés de différentes manières : il peut s'agir d'OpsItems parents et enfants ou partageant la même cause, ou encore de doublons. Vous pouvez associer un OpsItem à un autre pour l'afficher dans la section OpsItem connexes. Vous pouvez spécifier un maximum de 10 identifiants pour les autres OpsItems connexes à l'OpsItem actuel.



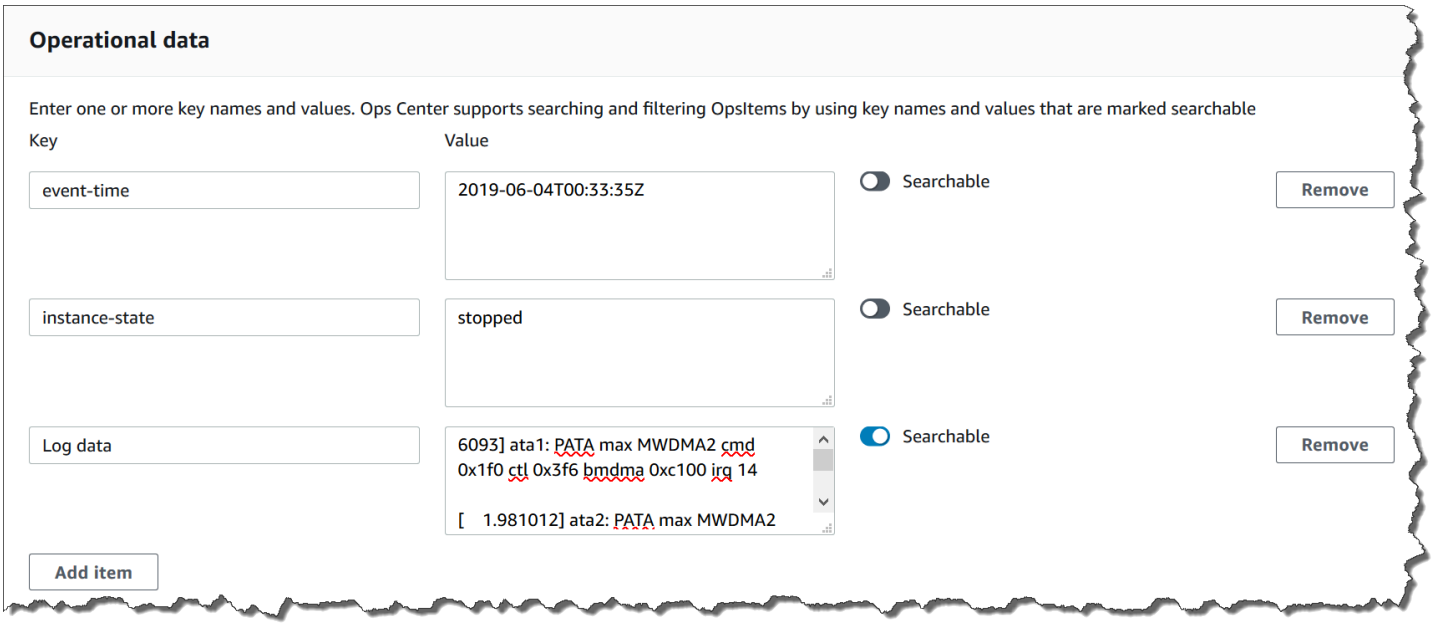
Pour ajouter un OpsItem associé

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez OpsCenter.
3. Sélectionnez un ID OpsItem pour ouvrir la page des détails.
4. Dans la section OpsItem connexes, sélectionnez Ajouter.
5. Pour OpsItem ID, spécifiez un identificateur.

6. Choisissez Add (Ajouter).

Ajout de données opérationnelles à un OpsItem

Les données opérationnelles sont des données personnalisées qui fournissent des références utiles sur un OpsItem. Vous pouvez saisir plusieurs paires clé/valeur pour les données opérationnelles. 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. La clé peut contenir jusqu'à 128 caractères et la taille de la valeur peut aller jusqu'à 20 Ko.



Vous pouvez rendre les données interrogeables par d'autres utilisateurs dans le compte ou, au contraire, limiter l'accès à la recherche. Les données sont dites interrogeables lorsque tous les utilisateurs autorisés à ouvrir la page Présentation d'un OpsItem (avec l'action d'API DescribeOpsItems) peuvent afficher les données spécifiées et faire des recherches dans celles-ci. Les données opérationnelles qui ne sont pas interrogeables sont uniquement visibles par les utilisateurs qui ont accès à l'OpsItem (comme fourni par l'opération 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, sélectionnez OpsCenter.
3. Sélectionnez un identifiant d'OpsItem pour ouvrir sa page de détails.

4. Développement des données opérationnelles.
5. S'il n'existe aucune donnée opérationnelle pour l'OpsItem, sélectionnez Ajouter. Si des données opérationnelles existent déjà pour le OpsItem, sélectionnez Gérer.

   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.
6. Pour Clé, spécifiez un ou plusieurs mots pour aider les utilisateurs à comprendre l'objectif des données.

   > **Important**
   >
   > Les clés de données opérationnelles ne peuvent pas commencer par : `amazon`, `aws`, `amzn`, `ssm`, `/amazon`, `/aws`, `/amzn` ou `/ssm`.
7. Dans Valeur, précisez les données.
8. Choisissez Enregistrer.

> **Note**
>
> Vous pouvez filtrer les OpsItems en utilisant l'opérateur Données opérationnelles sur la page OpsItems. Dans la zone Rechercher, sélectionnez 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"}`

Création d'un incident pour un OpsItem

Procédez comme suit pour créer un incident manuellement pour un OpsItem afin de le surveiller et de le gérer dans AWS Systems Manager Incident Manager, une fonctionnalité de AWS Systems Manager. Un incident est une interruption ou une réduction non planifiée de la qualité des services. Pour de plus amples informations sur Incident Manager, consultez the section called “Intégration des OpsCenter à d'autres Services AWS”.

Pour créer un incident manuellement pour 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, sélectionnez OpsCenter.
3. Si Incident Manager a créé un OpsItem pour vous, sélectionnez-le et passez à l'étape 5. Dans le cas contraire, sélectionnez Créer un OpsItem et remplissez le formulaire. Si vous ne voyez pas ce bouton, cliquez sur l'onglet OpsItems, puis sélectionnez Create OpsItem (Créer un OpsItem).
4. Si vous avez créé un OpsItem, ouvrez-le.
5. Sélectionnez Démarrer l'incident.
6. Pour Plan de réponse, sélectionnez le plan de réponse Incident Manager que vous souhaitez affecter à cet incident.
7. (Facultatif) Pour Titre, saisissez un nom descriptif qui aidera les autres membres de l'équipe à comprendre la nature de l'incident. Si vous ne spécifiez pas de nouveau titre, OpsCenter crée l'OpsItem et l'incident correspondant dans Incident Manager en utilisant le titre dans le plan de réponse.
8. (Facultatif) Pour Impact de l'incident, sélectionnez un niveau d'impact pour cet incident. Si vous ne sélectionnez pas de niveau d'impact, OpsCenter crée l'OpsItem et l'incident correspondant dans Incident Manager en utilisant le niveau d'impact dans le plan de réponse.
9. Sélectionnez Démarrer.

Gestion des OpsItems en double

OpsCenter peut recevoir plusieurs OpsItems en double de différents Services AWS pour une même source. OpsCenter a recours à une logique intégrée combinée à des chaînes de déduplication configurables pour éviter la création d'OpsItems en double. AWS Systems Manager applique une logique de déduplication intégrée lorsque l'opération Créer une API OpsItem est appelée.

AWS Systems Manager a recours à la logique de déduplication suivante :

1. 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 lancé l'OpsItem.
2. Lorsqu'une autre demande est faite pour créer un OpsItem, le système vérifie la chaîne de déduplication de la nouvelle demande.

3. Si un hachage correspondant existe pour cette chaîne de déduplication, Systems Manager vérifie le statut de l'OpsItem existant. Si le statut d'un OpsItem existant est ouvert ou en cours, l'OpsItem n'est pas créé. Si l'OpsItem existant est résolu, Systems Manager crée un nouvel OpsItem.

Une fois que vous avez créé un OpsItem, vous ne pouvez pas modifier ni changer les chaînes de déduplication dans cet OpsItem.

Pour gérer les OpsItems en double, vous pouvez effectuer les opérations suivantes :

- Modifiez la chaîne de déduplication pour une règle Amazon EventBridge ciblant OpsCenter. Pour de plus amples informations, consultez Modification d'une chaîne de déduplication dans une règle par défaut d'EventBridge.
- Vous pouvez spécifier une chaîne de déduplication lorsque vous créez manuellement un OpsItem. Pour de plus amples informations, veuillez consulter Spécification d'une chaîne de déduplication à l'aide de l'AWS CLI.
- Passez en revue et résolvez les OpsItems en double à l'aide d'informations opérationnelles. Vous pouvez utiliser des runbooks pour résoudre les OpsItems en double.

  Pour vous aider à résoudre les problèmes d'OpsItems en double et à réduire le nombre d'OpsItems créés par source, Systems Manager met à votre disposition les runbooks Automation suivants. Pour plus d'informations, consultez Résolution d'OpsItems en double sur la base d'informations.

Modification d'une chaîne de déduplication dans une règle par défaut d'EventBridge

Utilisez la procédure suivante pour spécifier une chaîne de déduplication pour une règle EventBridge qui cible OpsCenter.

Pour modifier une chaîne de déduplication dans une règle d'EventBridge, procédez comme suit :

1. Connectez-vous à la AWS Management Console et ouvrez la console Amazon EventBridge à l'adresse https://console.aws.amazon.com/events/.
2. Dans le volet de navigation, sélectionnez Rules.
3. Sélectionnez la règle, puis Edit (Modifier).
4. Accédez à la page Select target(s) (Sélectionner une ou plusieurs cibles).
5. Dans la section Additional settings (Réglages supplémentaires), choisissez Configure input transformer (Configurer le transformateur d'entrée).

6. Dans la case Template (Modèle), recherchez l'entrée JSON `"operationalData": { "/aws/dedup"` et les 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.

   ```
   "operationalData": { "/aws/dedup": {"type": "SearchableString","value":
    "{\"dedupString\":\"SSMOpsCenter-EBS-volume-performance-issue\"}"}}
   ```

7. Modifiez les chaînes de déduplication, puis choisissez Confirmer.
8. Choisissez Next (Suivant).
9. Choisissez Next (Suivant).
10. Choisissez Mettre à jour 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 à l'aide de la console AWS Systems Manager ou de l'AWS CLI. Pour plus d'informations sur la saisie des chaînes de déduplication lorsque vous créez manuellement un OpsItem dans la console, consultez Créer manuellement OpsItems. Si vous utilisez l'AWS CLI, vous pouvez indiquer la chaîne de déduplication pour le paramètre `OperationalData`. La syntaxe des paramètres utilise le format JSON, comme le montre l'exemple suivant.

```
--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 & macOS

```
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"
```

Analyse des informations opérationnelles pour réduire OpsItems

Les informations opérationnelles OpsCenter affichent des informations sur les OpsItems en double. OpsCenteranalyse automatiquement OpsItems dans votre compte et génère trois types d'informations. Vous pouvez consulter ces informations dans la section Informations opérationnelles de l'onglet OpsCenter Récapitulatif.

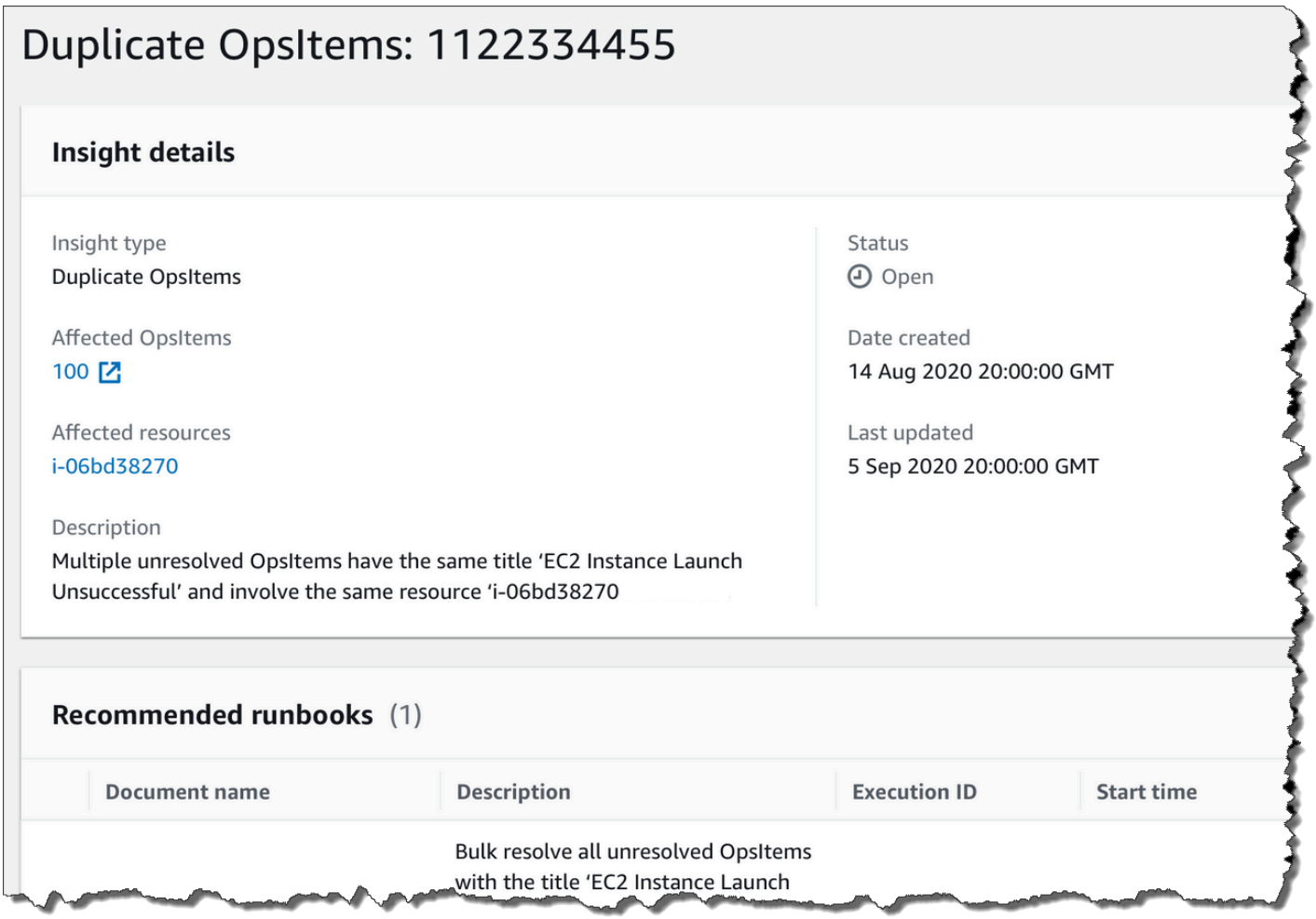
- OpsItems en double – Une information est générée lorsque huit OpsItems ou plus présentent le même titre pour la même ressource.
- Titres les plus courants – Une information est générée lorsque plus de 50 OpsItems présentent le même titre.
- Ressources générant le plus d'OpsItems – Une information est générée lorsqu'une ressource AWS en a plus de 10 OpsItems ouvertes. Ces informations et les ressources correspondantes sont affichées dans le tableau Ressources générant le plus d'OpsItems de l'onglet OpsCenter Récapitulatif. Les ressources sont répertoriées par ordre décroissant du nombre d'OpsItem.

> **Note**
>
> OpsCenter crée les Ressources générant le plus d'informations OpsItems pour les types de ressources suivants :

- Instances Amazon Elastic Compute Cloud (Amazon EC2)
- Groupes de sécurité Amazon EC2
- Groupe Amazon EC2 Auto Scaling
- Base de données Amazon Relational Database Service (Amazon RDS)
- Cluster Amazon RDS
- Fonction AWS Lambda
- Table Amazon DynamoDB
- Équilibreur de charge Elastic Load Balancing
- Cluster Amazon Redshift
- Certificat AWS Certificate Manager
- Volume Amazon Elastic Block Store

OpsCenter impose une limite de 15 informations par type. Si un type atteint cette limite, OpsCenter cesse d'afficher plus d'informations pour ce type. Pour obtenir des informations supplémentaires, vous devez résoudre tous les OpsItems associés à un OpsInsight de ce type. Si une information en attente ne peut pas être affichée dans la console en raison de la limite de 15 informations, cette information devient visible après la fermeture d'une autre information.

Quand vous sélectionnez une information, OpsCenter affiche les données liées aux OpsItems et aux ressources concernés. La capture d'écran suivante illustre un exemple avec les détails d'une information OpsItem en double.



Les informations opérationnelles sont désactivées par défaut. Pour plus d'informations sur l'utilisation des informations opérationnelles, consultez les rubriques suivantes.

Rubriques

- Activation des informations opérationnelles
- Résolution d'OpsItems en double sur la base d'informations
- Désactivation des informations opérationnelles

Activation des informations opérationnelles

Vous pouvez activer les informations opérationnelles sur la page OpsCenter de la console Systems Manager. Quand vous activez les informations opérationnelles, Systems Manager crée un rôle AWS Identity and Access Management (IAM) lié au service, appelé `AWSServiceRoleForAmazonSSM_OpsInsights`. 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 et incluent toutes les autorisations requises par le service pour appeler d'autres Services AWS en votre nom. Pour de plus amples informations sur le rôle lié à un service `AWSServiceRoleForAmazonSSM_OpsInsights`, consultez L'utilisation des rôles pour la création d'informations opérationnelles d'OpsItems dans l'OpsCenter de Systems Manager.

> **Note**
>
> Notez les informations importantes suivantes :
>
> - Votre Compte AWS est facturé pour les informations opérationnelles. Pour plus d'informations, consultez AWS Systems Manager Pricing (Tarification CTlong).
> - OpsCenter actualise périodiquement les informations à l'aide d'un traitement par lots. Cela signifie que la liste des informations affichées dans OpsCenter pourrait être désynchronisée.

Procédez comme suit pour activer et afficher les informations opérationnelles dans OpsCenter.

Pour activer et afficher des informations opérationnelles, 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, sélectionnez OpsCenter.
3. Dans la boîte de dialogue Les informations opérationnelles sont disponibles, choisissez Activer. Si ce message ne s'affiche pas, faites défiler l'écran jusqu'à la section Informations opérationnelles et choisissez Activer.
4. Après avoir activé cette fonctionnalité, dans l'onglet Récapitulatif, faites défiler l'écran vers le bas jusqu'à la section Informations opérationnelles.
5. Pour afficher une liste filtrée d'informations, choisissez le lien en regard de Dupliquer OpsItems, Titres les plus courants ou Ressources générant le plus d'OpsItems. Pour afficher toutes les informations, sélectionnez Afficher toutes les informations opérationnelles.
6. Sélectionnez un ID d'informations pour afficher de plus amples informations.

Résolution d'OpsItems en double sur la base d'informations

Pour résoudre les informations, vous devez d'abord résoudre tous les OpsItems associés à une information. Vous pouvez utiliser le runbook `AWS-BulkResolveOpsItemsForInsight` pour résoudre des OpsItems associés à une information.

Pour vous aider à résoudre les problèmes d'OpsItems en double et à réduire le nombre d'OpsItems créés par source, Systems Manager met à votre disposition les runbooks Automation suivants :

- Le runbook `AWS-BulkResolveOpsItems` résout des OpsItems correspondant à un filtre spécifié.
- Le runbook `AWS-AddOpsItemDedupStringToEventBridgeRule` ajoute une chaîne de déduplication pour toutes les cibles OpsItem associées à une règle Amazon EventBridge spécifique. Ce runbook n'ajoute pas de chaîne de déduplication si une règle en comprend déjà une.
- L'`AWS-DisableEventBridgeRule` désactive une règle dans EventBridge si celle-ci génère plusieurs dizaines ou centaines d'OpsItems.

Résolution des informations opérationnelles

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez OpsCenter.
3. Sur la page Présentation, faites défiler vers le bas jusqu'à Informations opérationnelles.
4. Choisissez Afficher toutes les informations opérationnelles.
5. Sélectionnez un ID d'informations pour afficher de plus amples informations.
6. Sélectionnez un runbook, puis sélectionnez Exécuter.

Désactivation des informations opérationnelles

Lorsque vous désactivez les informations opérationnelles, le système cesse de créer de nouvelles informations et de les afficher dans la console. Toutes les informations actives restent inchangées dans le système, bien qu'elles ne s'affichent pas dans la console. Si vous réactivez cette fonction, le système affiche les informations non résolues précédemment et lance la création de nouvelles informations. Procédez comme suit pour désactiver les informations opérationnelles.

Pour désactiver les informations opérationnelles, 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, sélectionnez OpsCenter.
3. Sélectionnez Settings (Paramètres).
4. Dans la section Informations opérationnelles, sélectionnez Modifier, puis basculez l'option Désactiver.
5. Choisissez Enregistrer.

Affichage des journaux et des rapports OpsCenter

AWS CloudTrail journalise les appels d'API AWS Systems Manager OpsCenter vers la console, l'AWS Command Line Interface (AWS CLI) et le SDK. Vous pouvez afficher ces informations dans la console CloudTrail ou dans un compartiment Amazon Simple Storage Service (Amazon S3). Un seul compartiment Amazon S3 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, consultez Journalisation des appels d' AWS Systems Manager API avec AWS CloudTrail.

AWS Systems Manager OpsCenter fournit les informations suivantes sur les OpsItems :

- Résumé de statut de l'OpsItem – Fournit un récapitulatif de l'OpsItems par état (Ouvert, En cours, Résolu, Ouvert et En cours).
- Sources avec la plupart des OpsItems ouverts – Fournit une répartition des principaux Services AWS qui comportent un OpsItems ouvert.
- OpsItems par source et âge – Fournit un nombre d'OpsItems regroupés par source et jours depuis la création.

Pour afficher le rapport récapitulatif dans OpsCenter, 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, sélectionnez OpsCenter.

3. Sur la page Présentation d'OpsItems, sélectionnez Récapitulatif.
4. Sous OpsItems par source et âge, sélectionnez la barre de recherche pour filtrer les OpsItems en fonction de la Source. Utilisez la liste pour filtrer en fonction de l'état.

Supprimez OpsItems

Vous pouvez supprimer un OpsItem individuel en appelant l'opération API DeleteOpsItem à l'aide du kit SDK AWS Command Line Interface ou AWS. Vous ne pouvez pas supprimer un OpsItem dans la AWS Management Console. Pour supprimer un OpsItem, votre utilisateur, groupe ou rôle AWS Identity and Access Management (IAM) doit disposer de l'autorisation d'administrateur ou vous devez avoir reçu l'autorisation d'appeler l'opération API `DeleteOpsItem`.

Important

Veuillez tenir compte des informations importantes suivantes relatives à cette opération.

- La suppression d'un OpsItem est irréversible. Il n'est pas possible de récupérer un OpsItem supprimé.
- Cette opération utilise un modèle de cohérence éventuelle, ce qui signifie que le système peut prendre quelques minutes pour terminer cette opération. Si vous supprimez un OpsItem et immédiatement appelez, par exemple GetOpsItem, le OpsItem supprimé peut toujours apparaître dans la réponse.
- Cette opération est idempotente. Le système ne génère pas d'exception si vous appelez cette opération plusieurs fois pour le même OpsItem. Si le premier appel est réussi, tous les appels supplémentaires renvoient la même réponse réussie que le premier appel.
- Cette opération ne prend pas en charge les appels entre comptes. Un compte d'administrateur délégué ou un compte de gestion ne peut pas supprimer des OpsItems dans d'autres comptes, même si OpsCenter est configuré pour l'administration entre comptes. Pour plus d'informations sur l'administration entre comptes, consultez (Facultatif) Configuration d'OpsCenter pour gérer de manière centralisée les OpsItems de l'ensemble des comptes (français non garanti).
- Si vous recevez le `OpsItemLimitExceededException`, vous pouvez supprimer un ou plusieurs OpsItems afin de réduire le nombre total de OpsItems en dessous des limites du quota. Pour plus d'informations sur cette exception, consultez Résolution des problèmes liés à OpsCenter (français non garanti).

Suppression d'un OpsItem

Utilisez la procédure suivante pour supprimer un OpsItem.

Pour supprimer un OpsItem

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS CLI. Pour plus d'informations, consultez Installation ou mise à jour de la version la plus récente de l'AWS CLI.
2. Exécutez la commande suivante. Remplacez *ID* par l'ID de la OpsItem que vous souhaitez supprimer.

```
aws ssm delete-ops-item --OpsItemId ID
```

En cas de réussite, la commande ne renvoie aucune donnée.

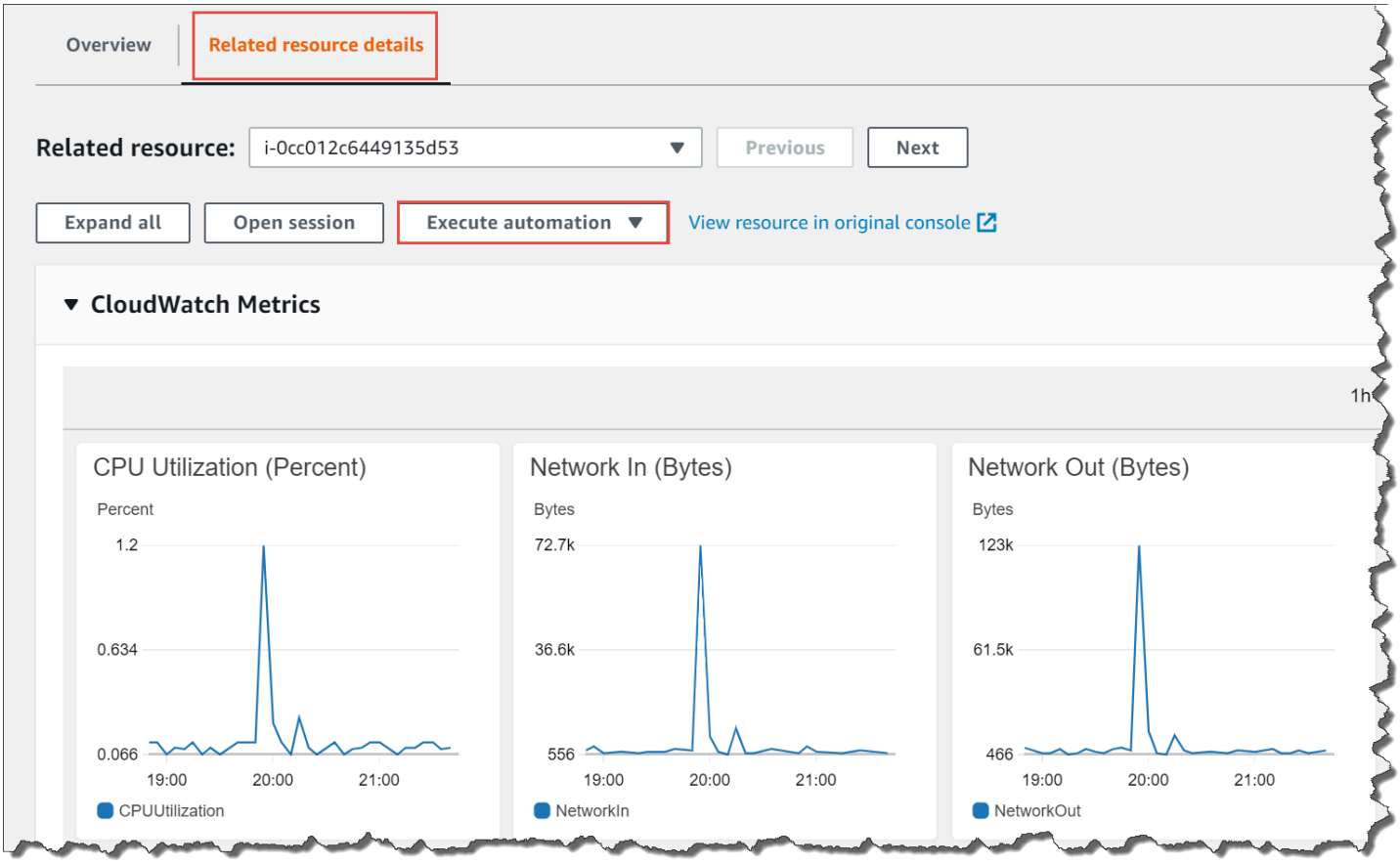
Correction des problèmes d'OpsItem

À l'aide des runbooks d' AWS Systems Manager automatisation, vous pouvez résoudre les problèmes liés aux AWS ressources identifiées dans un. OpsItem L'automatisation utilise des runbooks prédéfinis pour résoudre les problèmes courants liés aux AWS ressources.

Chaque OpsItem inclut la section Runbooks qui fournit une liste de runbooks que vous pouvez utiliser à des fins de correction. Lorsque vous sélectionnez un runbook Automation dans la liste, OpsCenter préremplit certains des champs requis pour exécuter le document. Lorsque vous exécutez un runbook Automation, le système associe le runbook à la ressource associée de l'OpsItem. Si Amazon EventBridge crée unOpsItem, il associe un runbook auOpsItem. OpsCenterconserve un enregistrement de 30 jours des runbooks d'automatisation pour un. OpsItem

Vous pouvez choisir un statut pour afficher des détails importants sur ce runbook, tels que la raison pour laquelle l'automatisation a échoué et l'étape du runbook Automation qui s'exécutait au moment de l'échec, comme le montre l'exemple suivant.

La page Détails relatifs à des ressources pour un OpsItem sélectionné inclut la liste des Exécutions d'Automation. Vous pouvez choisir des runbooks Automation récents ou spécifiques aux ressources, et les exécuter pour résoudre les problèmes. Cette page inclut également les fournisseurs de données, notamment CloudWatch les métriques et les alarmes Amazon, AWS CloudTrail les journaux et les informations provenant de AWS Config.



Vous pouvez afficher les informations sur un runbook Automation en choisissant le nom de runbook dans la console ou à l'aide de Référence du runbook Systems Manager Automation.

Corriger un OpsItem à l'aide d'un runbook

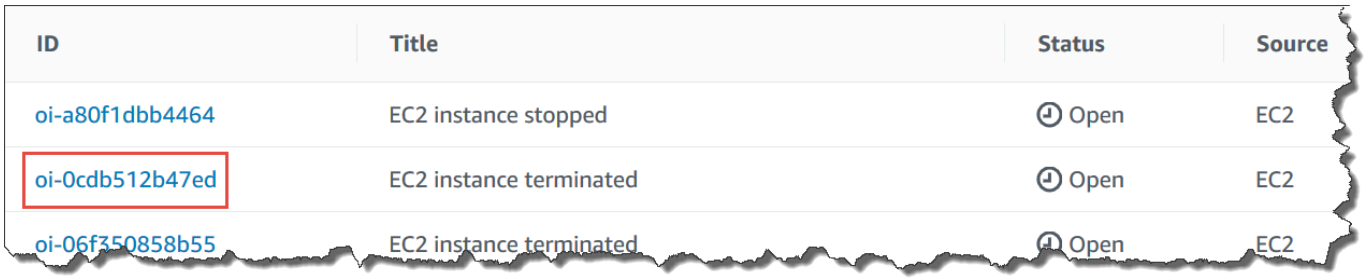
Avant d'exécuter un runbook Automation pour corriger un problème d'OpsItem, procédez comme suit :

- Vérifiez que vous avez l'autorisation d'exécuter des runbooks Systems Manager Automation. Pour plus d'informations, consultez Configuration d'Automation.
- Collectez des informations d'ID de ressource spécifique pour l'automatisation que vous souhaitez exécuter. Par exemple, si vous voulez exécuter une automatisation qui redémarre une instance EC2, vous devez spécifier l'ID de l'instance à redémarrer.

Pour exécuter un runbook Automation afin de corriger un problème d'OpsItem

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.

2. Dans le panneau de navigation, sélectionnez OpsCenter.
3. Sélectionnez l'ID OpsItem pour ouvrir la page des détails.



4. Faites défiler jusqu'à la section Runbooks.
5. Utilisez la barre de recherche ou les nombres dans le coin supérieur droit pour trouver les runbooks Automation que vous souhaitez exécuter.
6. Sélectionnez un runbook, puis sélectionnez Exécuter.
7. Saisissez les informations requises pour le runbook, puis sélectionnez Envoyer.

   Une fois le runbook lancé, le système revient à l'écran précédent et affiche le statut.
8. Dans la section Exécutions automatiques au cours des 30 derniers jours, sélectionnez le lien ID d'exécution pour afficher les étapes et le statut de l'exécution.

Corriger un OpsItem à l'aide d'un runbook associé

Après avoir exécuté un runbook Automation à partir d'un OpsItem, OpsCenter associe le runbook à l'OpsItem. Les runbooks associés ont un niveau supérieur aux autres dans la liste Runbooks.

Utilisez la procédure suivante pour exécuter un runbook Automation qui a déjà été associé à une ressource OpsItem. Pour plus d'informations sur l'ajout de ressources connexes, consultez Gestion des OpsItems.

Pour exécuter un runbook associé à une ressource afin de corriger un problème OpsItem

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez OpsCenter.
3. Ouvrez la OpsItem.
4. Dans la section Ressources connexes, sélectionnez la ressource sur laquelle vous voulez exécuter un runbook Automation.

5. Sélectionnez Run automation (Exécuter l'automatisation), puis le runbook Automation associé que vous souhaitez exécuter.
6. Entrez les informations requises pour le runbook, puis sélectionnez Exécuter.

   Une fois le runbook lancé, le système revient à l'écran précédent et affiche le statut.
7. Dans la section Exécutions automatiques au cours des 30 derniers jours, sélectionnez le lien ID d'exécution pour afficher les étapes et le statut de l'exécution.

Affichage des rapports de synthèse OpsCenter

OpsCenter AWS Systems Manager inclut une page récapitulative qui affiche automatiquement les informations suivantes :

- Résumé du statut de OpsItem : un résumé de OpsItems par statut, tel que `Open` et `In progress`.
- Sources avec la plupart des OpsItems ouverts : une répartition des principaux Services AWS qui comportent un OpsItems ouvert.
- OpsItems par source et âge : un nombre d'OpsItems regroupés par source et jours depuis la création.

Pour afficher les rapports de synthèse 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, puis l'onglet Résumé.
3. Dans la section OpsItems par source et par âge, effectuez les opérations suivantes :
   1. (Facultatif) Dans le champ de filtre, choisissez Source, sélectionnez `Equal`, `Begin With` ou `Not Equal`, puis saisissez un paramètre de recherche.
   2. Dans la liste adjacente, sélectionnez l'une des valeurs de statut suivantes :
      - `Open`
      - `In progress`
      - `Resolved`
      - `Open and in progress`
      - `All`

Résolution des problèmes liés à OpsCenter

Cette rubrique comprend des informations qui vous aideront à résoudre les erreurs et problèmes courants liés à OpsCenter.

OpsItemLimitExceededException s'affiche

Si votre Compte AWS a atteint le nombre maximum autorisé de OpsItems lorsque vous appelez l'opération d'API CreateOpsItem, `OpsItemLimitExceededException` s'affiche. OpsCenter renvoie l'exception si votre appel dépasse le nombre maximum de OpsItems pour l'un des quotas suivants :

- Nombre total de OpsItems par Compte AWS par région (y compris les OpsItems `Open` et `Resolved`) : 500 000
- Nombre maximal de OpsItems par Compte AWS par mois : 10 000

Ces quotas s'appliquent aux OpsItems créés à partir de n'importe quelle source, à l'exception des sources suivantes :

- OpsItems créé par les résultats du AWS Security Hub
- OpsItems qui sont générés automatiquement lors de l'ouverture d'un incident d'Incident Manager

Les OpsItems créés à partir de ces sources ne sont pas prises en compte dans vos quotas de OpsItem, mais vous êtes facturé pour chaque OpsItem.

Si vous recevez un `OpsItemLimitExceededException`, vous pouvez supprimer manuellement OpsItems jusqu'à ce que vous soyez en dessous du quota qui vous empêche de créer un nouveau OpsItem. Encore une fois, la suppression des OpsItems créés pour les résultats de Security Hub ou les incidents d'Incident Manager ne réduira pas le nombre total de OpsItems appliquées par les quotas. Vous devez supprimer les OpsItems des autres sources. Pour plus d'informations sur la suppression d'un OpsItem, consultez Supprimez OpsItems (français non garanti).

Vous recevez une facture élevée de la part d'AWS pour un grand nombre de OpsItems générés automatiquement

Si vous avez configuré l'intégration avec AWS Security Hub, OpsCenter crée des OpsItems pour les résultats de Security Hub. En fonction du nombre de résultats générées par Security Hub et

du compte auquel vous étiez connecté lorsque vous avez configuré l'intégration, OpsCenter peut générer un grand nombre de OpsItems, moyennant un coût. Voici des informations plus spécifiques relatives aux OpsItems générés par les résultats de Security Hub :

- Si vous êtes connecté au compte d'administrateur Security Hub lorsque vous configurez OpsCenter et l'intégration de Security Hub, le système crée des OpsItems pour les résultats dans le compte d'administrateur et dans tous les comptes membres. Les OpsItems sont tous créés dans le compte administrateur. En fonction de divers facteurs, cela peut entraîner une facture inhabituellement élevée de la part d'AWS.

  Si vous êtes connecté à un compte membre lorsque vous configurez l'intégration, le système crée uniquement des OpsItems pour les résultats de ce compte individuel. Pour plus d'informations sur le compte administrateur de Security Hub, les comptes membres et leur relation avec le flux d'événements EventBridge pour les résultats, consultez Types d'intégration de Security Hub avec EventBridge (français non garanti) dans le Guide de l'utilisateur AWS Security Hub.
- Pour chaque résultat qui crée un OpsItem, le prix normal de la création de l'OpsItem vous est facturé. Vous êtes également facturé si vous modifiez le OpsItem ou si le résultat correspondant est mis à jour dans Security Hub (ce qui déclenche une mise à jour du OpsItem).

> **Important**
>
> Si vous pensez qu'un grand nombre de OpsItems ont été créés par erreur et que votre facture AWS n'est pas justifiée, contactez AWS Support.

Suivez la procédure suivante si vous ne voulez plus que le système crée des OpsItems pour les résultats de Security Hub.

Pour ne plus recevoir des OpsItems pour les résultats de Security Hub

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez OpsCenter.
3. Sélectionnez Settings (Paramètres).
4. Dans la section Résultats de Security Hub, sélectionnez Modifier.
5. Cliquez sur le curseur pour faire passer l'option Activé à Désactivé. Si vous ne parvenez pas à basculer le curseur, Security Hub n'est pas activé pour votre Compte AWS.

6. Sélectionnez Enregistrer pour sauvegarder votre configuration. OpsCenter ne créera plus de OpsItems sur la base des résultats de Security Hub.

> **Important**
>
> Si OpsCenter renvoie le paramètre à Activé et continue la création de OpsItems pour les résultats, connectez-vous au compte d'administrateur délégué dans Systems Manager ou au compte de gestion AWS Organizations et répétez cette procédure. Si vous n'êtes pas autorisé à vous connecter à l'un de ces comptes, contactez votre administrateur et demandez-lui de répéter cette procédure pour désactiver l'intégration pour votre compte.

CloudWatchTableaux de bord Amazon hébergés par Systems Manager

Les CloudWatch tableaux de bord Amazon sont des pages d'accueil personnalisables dans la CloudWatch console que vous pouvez utiliser pour surveiller vos ressources dans une seule vue, même celles qui sont réparties sur différents Régions AWS sites. Vous pouvez utiliser CloudWatch des tableaux de bord pour créer des vues personnalisées des mesures et des alarmes relatives à vos AWS ressources. 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 AWS. Vous pouvez sélectionner la couleur utilisée pour chaque métrique sur chaque graphique afin de suivre 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, du AWS Command Line Interface (AWS CLI) ou de l' CloudWatch `PutDashboard`API. Pour plus d'informations, consultez la section Utilisation CloudWatch des tableaux de bord Amazon dans le guide de CloudWatch l'utilisateur Amazon.

AWS Systems Manager Gestion des applications

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

Rubriques

- AWS Systems Manager Application Manager
- AWS AppConfig
- AWS Systems Manager Parameter Store

AWS Systems Manager Application Manager

Application Manager, une fonctionnalité de AWS Systems Manager, aide les ingénieurs DevOps à étudier et résoudre les problèmes liés à leurs ressources AWS dans le contexte de leurs applications et clusters. Application Manager agrège des informations opérationnelles à partir de plusieurs Services AWS et fonctionnalités de Systems Manager sur une AWS Management Console unique.

Dans Application Manager, une application est un groupe logique de ressources AWS que vous voulez exploiter en tant qu'unité. Par exemple, ce groupe logique peut représenter différentes versions d'une application, ou les limites de propriété d'opérateurs ou d'environnements de développement. La prise en charge de clusters de conteneurs par Application Manager comprend à la fois les clusters Amazon Elastic Kubernetes Service (Amazon EKS) et Amazon Elastic Container Service (Amazon ECS).

Lorsque vous sélectionnez Mise en route sur la page d'accueil d'Application Manager, Application Manager importe automatiquement les métadonnées de vos ressources qui ont été créées dans d'autres Services AWS ou fonctionnalités de Systems Manager. Concernant les applications, Application Manager importe des métadonnées de l'ensemble de vos ressources AWS, organisées en groupes de ressources. Chaque groupe de ressources est répertorié dans la catégorie Applications personnalisées en tant qu'application unique. En outre, Application Manager importe automatiquement les métadonnées des ressources qui ont été créées par AWS CloudFormation, AWS Launch Wizard, Amazon ECS et Amazon EKS. Application Manager affiche ensuite ces ressources dans des catégories prédéfinies.

La liste des Applications inclut les éléments suivants :

- Applications personnalisées

- Launch Wizard
- Stacks CloudFormation
- Applications AppRegistry

La liste des Clusters de conteneurs inclut les éléments suivants :

- Clusters Amazon ECS
- Clusters Amazon EKS

Une fois l'importation terminée, vous pouvez afficher des informations opérationnelles sur vos ressources dans ces catégories prédéfinies. Autrement, si vous voulez ajouter un contexte à propos d'un ensemble de ressources, vous pouvez créer manuellement une application dans Application Manager et déplacer des ressources ou groupes de ressources vers cette application. Cela vous permet d'afficher des informations opérationnelles dans le contexte d'une application.

Après que vous avez paramétré et configuré des Services AWS et fonctionnalités de Systems Manager, Application Manager affiche les types d'informations suivants sur vos ressources :

- Informations sur l'état, le statut et l'intégrité d'Amazon EC2 Auto Scaling actuels des instances Amazon Elastic Compute Cloud (Amazon EC2) dans votre application
- Alarmes fournies par Amazon CloudWatch
- Informations sur la conformité fournies par AWS Config et State Manager (un composant de Systems Manager)
- Informations sur le cluster Kubernetes fournies par Amazon EKS
- Données de journalisation fournies par AWS CloudTrail et Amazon CloudWatch Logs
- OpsItems fournis par Systems Manager OpsCenter
- Détails des ressources fournies par Services AWS qui les hébergent.
- Informations sur le cluster de conteneurs fournies par Amazon ECS.

Pour vous aider à résoudre les problèmes liés à des composants ou ressources, Application Manager fournit également des runbooks que vous pouvez associer à vos applications. Pour vos premiers pas dans Application Manager, ouvrez Systems Manager console. Dans le panneau de navigation, sélectionnez Application Manager.

Quels sont les avantages liés à l'utilisation d'Application Manager ?

Application Manager réduit le temps nécessaire aux ingénieurs DevOps pour détecter et étudier les problèmes avec des ressources AWS. Pour cela, Application Manager affiche plusieurs types d'informations opérationnelles dans le contexte d'une application sur une seule console. Application Manager réduit également le temps nécessaire pour résoudre les problèmes en fournissant des runbooks qui effectuent des tâches de résolution courantes sur des ressources AWS.

Quelles sont les fonctions d'Application Manager ?

Application Manager inclut les fonctionnalités suivantes :

- Importer vos ressources AWS automatiquement

  Lors de la configuration initiale, vous pouvez choisir qu'Application Managerimporte et affiche automatiquement dans votre Compte AWS des ressources qui sont basées sur des piles CloudFormation, AWS Resource Groups, des déploiements Launch Wizard, des applications AppRegistry, et des clusters Amazon ECS et Amazon EKS. Le système affiche ces ressources dans des catégories prédéfinies d'applications ou de clusters. Par la suite, chaque fois que de nouvelles ressources de ces types sont ajoutées à votre Compte AWS, Application Manager les affiche automatiquement dans les catégories prédéfinies d'applications et de clusters.
- Créer ou modifier des piles et des modèles CloudFormation

  Application Manager vous aide à approvisionner et à gérer les ressources pour vos applications via l'intégration à CloudFormation. Vous pouvez créer, modifier et supprimer des modèles et des piles AWS CloudFormation dans Application Manager. Application Manager inclut également une bibliothèque de modèles dans laquelle vous pouvez cloner, créer et stocker des modèles. Application Manager et CloudFormation affichent les mêmes informations à propos du statut actuel d'une pile. Les modèles et les mises à jour de modèles sont stockés dans Systems Manager jusqu'à ce que vous approvisionniez la pile. Ensuite, les modifications s'affichent également dans CloudFormation.
- Afficher des informations sur vos instances dans le contexte d'une application

  Application Manager s'intègre à Amazon Elastic Compute Cloud (Amazon EC2) pour afficher des informations sur vos instances dans le contexte d'une application. Application Manager affiche l'état et le statut de l'instance et l'intégrité d'Amazon EC2 Auto Scaling pour une application sélectionnée dans un format graphique. L'onglet Instances inclut également un tableau contenant les informations suivantes pour chaque instance de votre application.

  - État de l'instance (Pending, Stopping, Running, Stopped [En attente, Arrêt, En cours d'exécution, Arrêtée])
  - Statut du ping de SSM Agent
  - Statut et nom du dernier runbook Systems Manager Automation traité sur l'instance
  - Nombre d'alertes Amazon CloudWatch Logs par état.
    - `ALARM` – La métrique ou l'expression se trouve à l'extérieur du seuil défini.
    - `OK` – La métrique ou l'expression se trouve dans le seuil défini.
    - `INSUFFICIENT_DATA` – L'alerte vient de commencer, la métrique n'est pas disponible, ou la quantité de données n'est pas suffisante pour permettre à la métrique de déterminer le statut de l'alerte.
  - Intégrité du groupe Auto Scaling pour les groupes de scalabilité automatique parent et individuel
- Afficher les métriques opérationnelles et les alarmes pour une application ou un cluster

  Application Manager s'intègre avec Amazon CloudWatch pour fournir des métriques opérationnelles et des alarmes en temps réel pour une application ou un cluster. Vous pouvez explorer votre arborescence d'applications pour afficher les alarmes au niveau de chaque composant ou d'un cluster individuel.
- Afficher les données de journalisation d'une application

  Application Manager s'intègre avec Amazon CloudWatch Logs pour fournir des données de journalisation dans le contexte de votre application, sans quitter Systems Manager.
- Afficher et gérer des OpsItems pour une application ou un cluster

  Application Manager s'intègre avec AWS Systems Manager OpsCenter pour fournir une liste d'éléments opérationnels (OpsItems) pour vos applications et clusters. La liste reflète les OpsItems générés automatiquement et ceux créés manuellement. Vous pouvez afficher des détails sur la ressource qui a créé un OpsItem, ainsi que le statut, la source et la sévérité de l'OpsItem.
- Afficher les données de conformité des ressources pour une application ou un cluster

  Application Manager s'intègre avec AWS Config pour fournir des détails sur la conformité et l'historique de vos ressources AWS selon les règles que vous spécifiez. Application Manager s'intègre également avec AWS Systems Manager State Manager pour fournir des informations de conformité relatives au statut que vous voulez maintenir pour vos instances Amazon Elastic Compute Cloud (Amazon EC2).
- Afficher les informations sur l'infrastructure de clusters Amazon ECS et Amazon EKS

Application Manager s'intègre avec Amazon ECS et Amazon EKS pour fournir des informations sur l'état de vos infrastructures de clusters, ainsi qu'une vue de l'environnement d'exécution des ressources de calcul, de mise en réseau et de stockage dans un cluster.

Il ne vous est toutefois pas possible de gérer ou d'afficher les informations opérationnelles relatives à vos pods ou conteneurs Amazon EKS dans Application Manager. Vous ne pouvez gérer et afficher que les informations opérationnelles relatives à l'infrastructure qui héberge vos ressources Amazon EKS.

- Afficher les détails liés au coût des ressources d'une application

  Application Manager est intégré à AWS Cost Explorer, une fonctionnalité de AWS Billing and Cost Management via le widget Cost. Une fois que vous avez activé l'explorateur de coûts dans la console de facturation et gestion des coûts, le widget Cost (Coût) dans Application Manager affiche les données de coût pour une application ou un composant d'application non-conteneurisés spécifiques. Vous pouvez appliquer des filtres dans le widget pour afficher les données de coût selon différentes périodes, différentes granularités et différents types sous forme de graphique à barres ou linéaire.

- Afficher des informations détaillées sur les ressources dans une console unique

  Sélectionnez un nom de ressource répertorié dans Application Manager, et affichez des informations contextuelles et des informations opérationnelles sur cette ressource sans quitter Systems Manager.

- Recevoir des mises à jour automatiques sur les ressources pour des applications

  Si vous apportez des modifications à une ressource dans une console de service et que cette ressource fait partie d'une application dans Application Manager, Systems Manager affiche automatiquement ces modifications. Par exemple, si vous mettez à jour une pile dans la console AWS CloudFormation et que cette pile fait partie d'une application dans Application Manager, les mises à jour de la pile sont automatiquement reflétées dans Application Manager.

- Découvrir automatiquement des applications de Launch Wizard

  Application Manager est intégré à AWS Launch Wizard. Si vous avez utilisé Launch Wizard pour déployer des ressources pour une application, Application Manager peut les importer et les afficher automatiquement dans une section de Launch Wizard.

- Surveiller des ressources d'application dans Application Manager en utilisant CloudWatch Application Insights

Application Manager s'intègre à Amazon CloudWatch Application Insights. Application Insights identifie et paramètre des métriques, des journaux et des alarmes clés dans vos ressources d'application et votre pile technologique. Application Insights surveille en permanence les métriques et les journaux afin de détecter et de corréler les anomalies et les erreurs. Lorsque des erreurs et des anomalies sont détectées, Application Insights génère des CloudWatch Events que vous pouvez utiliser pour paramétrer des notifications ou effectuer des actions. Vous pouvez activer et afficher Application Insights sous les onglets Overview (Présentation) et Monitoring (Surveillance) dans Application Manager. Pour de plus amples informations sur Application Insights, veuillez consulter Qu'est-ce qu'Amazon CloudWatch Application Insights dans le Guide de l'utilisateur Amazon CloudWatch.

- Résoudre des problèmes liés aux runbooks

  Application Manager inclut des runbooks Systems Manager prédéfinis pour résoudre les problèmes courants avec les ressources AWS. Vous pouvez exécuter un runbook sur toutes les ressources applicables d'une application sans avoir à quitter Application Manager.

L'utilisation d'Application Manager entraîne-t-elle des frais ?

Application Manager est disponible sans frais supplémentaires.

Quels sont les quotas de ressources pour Application Manager ?

Vous pouvez afficher des quotas pour toutes les fonctionnalités de Systems Manager dans la rubrique Quotas de service Systems Manager dans la Référence générale d'Amazon Web Services. Sauf indication contraire, chaque quota est spécifique à la région.

Rubriques

- Mise en route avec Systems Manager Application Manager
- Utilisation des Application Manager

Mise en route avec Systems Manager Application Manager

Utilisez les informations de cette section pour paramétrer et configurer Application Manager, une fonctionnalité de AWS Systems Manager, afin d'afficher les informations opérationnelles de différents Services AWS et fonctionnalités de Systems Manager. Cette section inclut également des informations sur l'ajout d'applications et de clusters à Application Manager.

Rubriques

Configuration des services connexes

Application Manager, une fonctionnalité de AWS Systems Manager, affiche des ressources et des informations provenant d'autres Services AWS et fonctionnalités de Systems Manager. Pour maximiser la quantité d'informations opérationnelles affichées dans Application Manager, nous vous recommandons de paramétrer et de configurer ces autres services ou fonctionnalités avant d'utiliser Application Manager.

Rubriques

Paramétrer des tâches pour l'importation de ressources

Les tâches de paramétrage suivantes vous aident à afficher des ressources AWS dans Application Manager. Lorsque toutes ces tâches sont terminées, Systems Manager peut importer automatiquement des ressources dans Application Manager. Une fois vos ressources importées, vous pouvez créer des applications dans Application Manager et y transférer vos ressources importées. Cela vous aide à afficher les informations opérationnelles dans le contexte d'une application.

(Facultatif) Organiser vos ressources AWS en utilisant des balises.

> Vous pouvez attribuer des métadonnées à vos ressources AWS sous la forme d'identifications. Chaque balise est une étiquette composée d'une clé définie par l'utilisateur et d'une valeur. Les balises peuvent vous aider à gérer, identifier, organiser, rechercher et filtrer des ressources. Vous pouvez créer des balises pour classer vos ressources par objectif, propriétaire, environnement ou selon d'autres critères.

(Facultatif) Organise vos ressources AWS en utilisant des balises AWS Resource Groups

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.

Application Manager importe automatiquement tous vos groupes de ressources et les répertorie dans la catégorie Applications personnalisées.

(Facultatif) Paramétrer et déployer vos ressources AWS en utilisant AWS CloudFormation

AWS CloudFormation – permet de créer et d'allouer des déploiements d'infrastructure AWS de manière prévisible et répétée. Ce service vous aide à utiliser des Services AWS comme Amazon EC2, Amazon Elastic Block Store (Amazon EBS), Amazon Simple Notification Service (Amazon SNS), Elastic Load Balancing et AWS Auto Scaling. Avec CloudFormation, vous pouvez créer des applications fiables, évolutives et rentables dans le cloud sans vous soucier de créer et de configurer l'infrastructure AWS sous-jacente.

Application Manager importe automatiquement tous vos ressources AWS CloudFormation et les répertorie dans la catégorie Piles AWS CloudFormation. Vous pouvez créer des piles et des modèles CloudFormation dans Application Manager. Les modifications de piles et de modèles sont automatiquement synchronisées entre Application Manager et CloudFormation. Vous pouvez également créer des applications dans Application Manager et y transférer des piles. Cela vous aide à afficher les informations opérationnelles de ressources dans vos piles, dans le contexte d'une application. Pour de plus amples informations sur la tarification, veuillez consulter Tarification de AWS CloudFormation.

(Facultatif) Paramétrer et déployer vos applications en utilisant AWS Launch Wizard

Launch Wizard vous guide tout au long du processus de dimensionnement, de configuration et de déploiement de ressources AWS pour des applications tierces, sans la nécessité d'identifier et d'approvisionner manuellement des ressources AWS.

Application Manager importe automatiquement toutes vos ressources Launch Wizard et les répertorie dans la catégorie Launch Wizard. Pour de plus amples informations sur AWS Launch Wizard, veuillez consulter Mise en route avec AWS Launch Wizard pour SQL Server. Launch Wizard est disponible sans frais supplémentaires. Vous ne payez que les ressources AWS que vous approvisionnez pour exécuter votre solution.

(Facultatif) Paramétrer et déployer vos applications conteneurisées en utilisant Amazon ECS et Amazon EKS

Amazon Elastic Container Service (Amazon ECS) est un service de gestion de conteneurs hautement évolutif et rapide, qui facilite l'exécution, l'arrêt et la gestion de conteneurs Docker sur un cluster. Vos conteneurs sont définis dans une définition de tâche qui vous sert à exécuter des tâches individuelles ou des tâches dans un service.

Amazon EKS est un service géré que vous pouvez utiliser pour exécuter Kubernetes sur AWS sans avoir à installer, à utiliser et à entretenir votre propre plan de contrôle Kubernetes ou nœuds. Kubernetes est un système open source destiné à l'automatisation du déploiement, la mise à l'échelle et la gestion d'applications conteneurisées.

Application Manager importe automatiquement toutes vos ressources d'infrastructure Amazon ECS et Amazon EKS, et les répertorie sous l'onglet Clusters de conteneurs. Il ne vous est toutefois pas possible de gérer ou d'afficher les informations opérationnelles relatives à vos pods ou conteneurs Amazon EKS dans Application Manager. Vous ne pouvez gérer et afficher que les informations opérationnelles relatives à l'infrastructure qui héberge vos ressources Amazon EKS. Pour plus d'informations, consultez Tarification Amazon ECS et Tarification Amazon EKS.

Paramétrer des tâches pour afficher des informations opérationnelles relatives aux ressources

Les tâches de paramétrage suivantes vous aident à afficher les informations opérationnelles relatives à vos ressources AWS dans Application Manager.

(Recommandé) Vérifier les autorisations de runbook

Vous pouvez résoudre les problèmes avec les ressources AWS à partir d'Application Manager en utilisant des runbooks Systems Manager Automation. Pour utiliser cette fonctionnalité de résolution, vous devez configurer ou vérifier les autorisations. Pour de plus amples informations sur la tarification, veuillez consulter Tarification de AWS Systems Manager.

(Facultatif) Activer Cost Explorer

AWS Cost Explorer est une fonction de AWS Cost Management que vous pouvez utiliser pour visualiser vos données de coûts en vue d'une analyse plus approfondie. Lorsque vous activez Cost Explorer, vous pouvez afficher les informations sur les coûts, l'historique des coûts et l'optimisation des coûts pour les ressources de votre application dans la console Application Manager.

(Facultatif) Paramétrer et configurer Amazon CloudWatch Logs et les alarmes

CloudWatch est un service de surveillance et de gestion qui fournit des données et des informations exploitables pour les applications AWS, hybrides et multicloud, et les ressources d'infrastructure. Avec CloudWatch, vous pouvez collecter et accéder à toutes vos données opérationnelles et de performance sous forme de journaux et de métriques à partir d'une seule plateforme. Pour afficher CloudWatch Logs et les alarmes pour vos ressources dans Application Manager, vous devez paramétrer et configurer CloudWatch. Pour de plus amples informations, veuillez consulter Tarification CloudWatch.

> **Note**
>
> La prise en charge de CloudWatch Logs s'applique uniquement aux applications, mais pas aux clusters.

(Facultatif) Paramétrer et configurer AWS Config

AWS Config offre une vue détaillée des ressources associées à votre Compte AWS, notamment la façon dont elles sont configurées, leurs relations mutuelles et l'évolution dans le temps des configurations et de leurs relations. Vous pouvez utiliser AWS Config pour évaluer les paramètres de configuration de vos ressources AWS. Pour cela, vous devez créer des règles AWS Config représentatives de vos paramètres de configuration idéaux. Pendant qu'AWS Config surveille en permanence les changements de configuration de vos ressources, il vérifie si ces changements ne vont pas à l'encontre de l'une des conditions de vos règles. Si une ressource va à l'encontre d'une règle, AWS Config signale la ressource et la règle comme non conformes. Application Manager affiche des informations de conformité à propos des règles AWS Config. Pour afficher ces données dans Application Manager, vous devez installer et configurer AWS Config. Pour de plus amples informations sur la tarification, veuillez consulter Tarification de AWS Config.

(Facultatif) Créer des associations State Manager

Vous pouvez utiliser Systems Manager State Manager pour créer la configuration à affecter à vos nœuds gérés. Cette configuration, appelée association, définit l'état que vous souhaitez maintenir sur vos nœuds. Pour afficher les données de conformité des associations dans Application Manager, vous devez configurer une ou plusieurs associations State Manager. State Manager est offert gratuitement.

(Facultatif) Paramétrer et configurer OpsCenter

Vous pouvez afficher les éléments opérationnels (OpsItems) relatifs à vos ressources dans Application Manager en utilisant OpsCenter. Vous pouvez configurer Amazon CloudWatch et Amazon EventBridge pour l'envoi automatique d'OpsItems à OpsCenter en fonction des alarmes et des événements. Vous pouvez également saisir OpsItems manuellement. Pour de plus amples informations sur la tarification, veuillez consulter Tarification de AWS Systems Manager.

Configuration des autorisations pour Systems Manager Application Manager

Vous pouvez utiliser toutes les fonctions d'Application Manager (fonctionnalité développée par AWS Systems Manager) si votre entité AWS Identity and Access Management (IAM) (qui peut être un utilisateur, un groupe ou un rôle) a accès aux opérations d'API répertoriées dans cette rubrique. Les opérations d'API sont divisées en deux tableaux pour vous aider à comprendre les différentes fonctions qu'elles exécutent.

Le tableau suivant répertorie les opérations d'API que Systems Manager appelle si vous sélectionnez une ressource dans Application Manager pour en afficher les détails. Par exemple, si Application Manager répertorie un groupe Amazon EC2 Auto Scaling et que vous sélectionnez ce groupe pour en afficher les détails, Systems Manager appelle les opérations d'API `autoscaling:DescribeAutoScalingGroups`. Si votre compte ne contient pas de groupes Auto Scaling, Application Manager n'appelle pas cette opération d'API.

Détails de ressource uniquement

```
acm:DescribeCertificate
acm:ListTagsForCertificate
autoscaling:DescribeAutoScalingGroups
cloudfront:GetDistribution
cloudfront:ListTagsForResource
cloudtrail:DescribeTrails
cloudtrail:ListTags
cloudtrail:LookupEvents
codebuild:BatchGetProjects
codepipeline:GetPipeline
codepipeline:ListTagsForResource
dynamodb:DescribeTable
dynamodb:ListTagsOfResource
ec2:DescribeAddresses
```

Détails de ressource uniquement

```
ec2:DescribeCustomerGateways
ec2:DescribeHosts
ec2:DescribeInternetGateways
ec2:DescribeNetworkAcls
ec2:DescribeNetworkInterfaces
ec2:DescribeRouteTables
ec2:DescribeSecurityGroups
ec2:DescribeSubnets
ec2:DescribeVolumes
ec2:DescribeVpcs
ec2:DescribeVpnConnections
ec2:DescribeVpnGateways
elasticbeanstalk:DescribeApplications
elasticbeanstalk:ListTagsForResource
elasticloadbalancing:DescribeInstanceHealth
elasticloadbalancing:DescribeListeners
elasticloadbalancing:DescribeLoadBalancers
elasticloadbalancing:DescribeTags
iam:GetGroup
iam:GetPolicy
iam:GetRole
iam:GetUser
lambda:GetFunction
rds:DescribeDBClusters
rds:DescribeDBInstances
rds:DescribeDBSecurityGroups
rds:DescribeDBSnapshots
rds:DescribeDBSubnetGroups
rds:DescribeEventSubscriptions
rds:ListTagsForResource
redshift:DescribeClusterParameters
redshift:DescribeClusterSecurityGroups
redshift:DescribeClusterSnapshots
redshift:DescribeClusterSubnetGroups
redshift:DescribeClusters
s3:GetBucketTagging
```

Le tableau suivant répertorie les opérations d'API utilisées par Systems Manager pour modifier les applications et les ressources répertoriées dans Application Manager ou pour afficher les informations opérationnelles d'une application ou d'une ressource sélectionnée.

Actions et détails d'une application

```
applicationinsights:CreateApplication
applicationinsights:DescribeApplication
applicationinsights:ListProblems
ce:GetCostAndUsage
ce:GetTags
ce:ListCostAllocationTags
ce:UpdateCostAllocationTagsStatus
cloudformation:CreateStack
cloudformation:DeleteStack
cloudformation:DescribeStackDriftDetectionStatus
cloudformation:DescribeStackEvents
cloudformation:DescribeStacks
cloudformation:DetectStackDrift
cloudformation:GetTemplate
cloudformation:GetTemplateSummary
cloudformation:ListStacks
cloudformation:UpdateStack
cloudwatch:DescribeAlarms
cloudwatch:DescribeInsightRules
cloudwatch:DisableAlarmActions
cloudwatch:EnableAlarmActions
cloudwatch:GetMetricData
cloudwatch:ListTagsForResource
cloudwatch:PutMetricAlarm
config:DescribeComplianceByConfigRule
config:DescribeComplianceByResource
config:DescribeConfigRules
config:DescribeRemediationConfigurations
config:GetComplianceDetailsByConfigRule
config:GetComplianceDetailsByResource
config:GetResourceConfigHistory
config:ListDiscoveredResources
config:PutRemediationConfigurations
config:SelectResourceConfig
config:StartConfigRulesEvaluation
config:StartRemediationExecution
ec2:DescribeInstances
ecs:DescribeCapacityProviders
ecs:DescribeClusters
ecs:DescribeContainerInstances
```

Actions et détails d'une application

```
ecs:ListClusters
ecs:ListContainerInstances
ecs:TagResource
eks:DescribeCluster
eks:DescribeFargateProfile
eks:DescribeNodegroup
eks:ListClusters
eks:ListFargateProfiles
eks:ListNodegroups
eks:TagResource
iam:CreateServiceLinkedRole
iam:ListRoles
logs:DescribeLogGroups
resource-groups:CreateGroup
resource-groups:DeleteGroup
resource-groups:GetGroup
resource-groups:GetGroupQuery
resource-groups:GetTags
resource-groups:ListGroupResources
resource-groups:ListGroups
resource-groups:Tag
resource-groups:Untag
resource-groups:UpdateGroup
s3:ListAllMyBuckets
s3:ListBucket
s3:ListBucketVersions
servicecatalog:GetApplication
servicecatalog:ListApplications
sns:CreateTopic
sns:ListSubscriptionsByTopic
sns:ListTopics
sns:Subscribe
ssm:AddTagsToResource
ssm:CreateDocument
ssm:CreateOpsMetadata
ssm:DeleteDocument
ssm:DeleteOpsMetadata
ssm:DescribeAssociation
ssm:DescribeAutomationExecutions
ssm:DescribeDocument
ssm:DescribeDocumentPermission
ssm:GetDocument
```

Actions et détails d'une application

```
ssm:GetInventory
ssm:GetOpsMetadata
ssm:GetOpsSummary
ssm:GetServiceSetting
ssm:ListAssociations
ssm:ListComplianceItems
ssm:ListDocuments
ssm:ListDocumentVersions
ssm:ListOpsMetadata
ssm:ListResourceComplianceSummaries
ssm:ListTagsForResource
ssm:ModifyDocumentPermission
ssm:RemoveTagsFromResource
ssm:StartAssociationsOnce
ssm:StartAutomationExecution
ssm:UpdateDocument
ssm:UpdateDocumentDefaultVersion
ssm:UpdateOpsItem
ssm:UpdateOpsMetadata
ssm:UpdateServiceSetting
tag:GetTagKeys
tag:GetTagValues
tag:TagResources
tag:UntagResources
```

Configuration des autorisations

Pour configurer les autorisations d'Application Manager pour une entité IAM (qui peut être un utilisateur, un groupe ou un rôle), créez une politique IAM à l'aide de l'exemple suivant. Cet exemple de politique inclut toutes les opérations d'API utilisées par Application Manager.

```
                {
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "acm:DescribeCertificate",
                "acm:ListTagsForCertificate",
```

```
                "applicationinsights:CreateApplication",
                "applicationinsights:DescribeApplication",
                "applicationinsights:ListProblems",
                "autoscaling:DescribeAutoScalingGroups",
                "ce:GetCostAndUsage",
                "ce:GetTags",
                "ce:ListCostAllocationTags",
                "ce:UpdateCostAllocationTagsStatus",
                "cloudformation:CreateStack",
                "cloudformation:DeleteStack",
                "cloudformation:DescribeStackDriftDetectionStatus",
                "cloudformation:DescribeStackEvents",
                "cloudformation:DescribeStacks",
                "cloudformation:DetectStackDrift",
                "cloudformation:GetTemplate",
                "cloudformation:GetTemplateSummary",
                "cloudformation:ListStacks",
                "cloudformation:ListStackResources",
                "cloudformation:UpdateStack",
                "cloudfront:GetDistribution",
                "cloudfront:ListTagsForResource",
                "cloudtrail:DescribeTrails",
                "cloudtrail:ListTags",
                "cloudtrail:LookupEvents",
                "cloudwatch:DescribeAlarms",
                "cloudwatch:DescribeInsightRules",
                "cloudwatch:DisableAlarmActions",
                "cloudwatch:EnableAlarmActions",
                "cloudwatch:GetMetricData",
                "cloudwatch:ListTagsForResource",
                "cloudwatch:PutMetricAlarm",
                "codebuild:BatchGetProjects",
                "codepipeline:GetPipeline",
                "codepipeline:ListTagsForResource",
                "config:DescribeComplianceByConfigRule",
                "config:DescribeComplianceByResource",
                "config:DescribeConfigRules",
                "config:DescribeRemediationConfigurations",
                "config:GetComplianceDetailsByConfigRule",
                "config:GetComplianceDetailsByResource",
                "config:GetResourceConfigHistory",
                "config:ListDiscoveredResources",
                "config:PutRemediationConfigurations",
                "config:SelectResourceConfig",
```

```
                "config:StartConfigRulesEvaluation",
                "config:StartRemediationExecution",
                "dynamodb:DescribeTable",
                "dynamodb:ListTagsOfResource",
                "ec2:DescribeAddresses",
                "ec2:DescribeCustomerGateways",
                "ec2:DescribeHosts",
                "ec2:DescribeInstances",
                "ec2:DescribeInternetGateways",
                "ec2:DescribeNetworkAcls",
                "ec2:DescribeNetworkInterfaces",
                "ec2:DescribeRouteTables",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeVolumes",
                "ec2:DescribeVpcs",
                "ec2:DescribeVpnConnections",
                "ec2:DescribeVpnGateways",
                "ecs:DescribeCapacityProviders",
                "ecs:DescribeClusters",
                "ecs:DescribeContainerInstances",
                "ecs:ListClusters",
                "ecs:ListContainerInstances",
                "ecs:TagResource",
                "eks:DescribeCluster",
                "eks:DescribeFargateProfile",
                "eks:DescribeNodegroup",
                "eks:ListClusters",
                "eks:ListFargateProfiles",
                "eks:ListNodegroups",
                "eks:TagResource",
                "elasticbeanstalk:DescribeApplications",
                "elasticbeanstalk:ListTagsForResource",
                "elasticloadbalancing:DescribeInstanceHealth",
                "elasticloadbalancing:DescribeListeners",
                "elasticloadbalancing:DescribeLoadBalancers",
                "elasticloadbalancing:DescribeTags",
                "iam:CreateServiceLinkedRole",
                "iam:GetGroup",
                "iam:GetPolicy",
                "iam:GetRole",
                "iam:GetUser",
                "iam:ListRoles",
                "lambda:GetFunction",
```

```
                "logs:DescribeLogGroups",
                "rds:DescribeDBClusters",
                "rds:DescribeDBInstances",
                "rds:DescribeDBSecurityGroups",
                "rds:DescribeDBSnapshots",
                "rds:DescribeDBSubnetGroups",
                "rds:DescribeEventSubscriptions",
                "rds:ListTagsForResource",
                "redshift:DescribeClusterParameters",
                "redshift:DescribeClusters",
                "redshift:DescribeClusterSecurityGroups",
                "redshift:DescribeClusterSnapshots",
                "redshift:DescribeClusterSubnetGroups",
                "resource-groups:CreateGroup",
                "resource-groups:DeleteGroup",
                "resource-groups:GetGroup",
                "resource-groups:GetGroupQuery",
                "resource-groups:GetTags",
                "resource-groups:ListGroupResources",
                "resource-groups:ListGroups",
                "resource-groups:Tag",
                "resource-groups:Untag",
                "resource-groups:UpdateGroup",
                "s3:GetBucketTagging",
                "s3:ListAllMyBuckets",
                "s3:ListBucket",
                "s3:ListBucketVersions",
                "servicecatalog:GetApplication",
                "servicecatalog:ListApplications",
                "sns:CreateTopic",
                "sns:ListSubscriptionsByTopic",
                "sns:ListTopics",
                "sns:Subscribe",
                "ssm:AddTagsToResource",
                "ssm:CreateDocument",
                "ssm:CreateOpsMetadata",
                "ssm:DeleteDocument",
                "ssm:DeleteOpsMetadata",
                "ssm:DescribeAssociation",
                "ssm:DescribeAutomationExecutions",
                "ssm:DescribeDocument",
                "ssm:DescribeDocumentPermission",
                "ssm:GetDocument",
                "ssm:GetInventory",
```

```
                "ssm:GetOpsMetadata",
                "ssm:GetOpsSummary",
                "ssm:GetServiceSetting",
                "ssm:ListAssociations",
                "ssm:ListComplianceItems",
                "ssm:ListDocuments",
                "ssm:ListDocumentVersions",
                "ssm:ListOpsMetadata",
                "ssm:ListResourceComplianceSummaries",
                "ssm:ListTagsForResource",
                "ssm:ModifyDocumentPermission",
                "ssm:RemoveTagsFromResource",
                "ssm:StartAssociationsOnce",
                "ssm:StartAutomationExecution",
                "ssm:UpdateDocument",
                "ssm:UpdateDocumentDefaultVersion",
                "ssm:UpdateOpsMetadata",
                "ssm:UpdateOpsItem",
                "ssm:UpdateServiceSetting",
                "tag:GetResources",
                "tag:GetTagKeys",
                "tag:GetTagValues",
                "tag:TagResources",
                "tag:UntagResources"
            ],
            "Resource": "*"
        }
    ]
}
```

Note

Vous pouvez restreindre la capacité d'un utilisateur à modifier des applications et des ressources dans Application Manager en supprimant les opérations d'API suivantes de la politique d'autorisations IAM attachée à leur utilisateur, groupe ou rôle. La suppression de ces actions crée une expérience en lecture seule dans Application Manager. Vous trouverez ci-dessous toutes les API qui permettent aux utilisateurs d'apporter des modifications à l'application ou à toute autre ressource associée.

```
applicationinsights:CreateApplication
ce:UpdateCostAllocationTagsStatus
```

```
cloudformation:CreateStack
cloudformation:DeleteStack
cloudformation:UpdateStack
cloudwatch:DisableAlarmActions
cloudwatch:EnableAlarmActions
cloudwatch:PutMetricAlarm
config:PutRemediationConfigurations
config:StartConfigRulesEvaluation
config:StartRemediationExecution
ecs:TagResource
eks:TagResource
iam:CreateServiceLinkedRole
resource-groups:CreateGroup
resource-groups:DeleteGroup
resource-groups:Tag
resource-groups:Untag
resource-groups:UpdateGroup
sns:CreateTopic
sns:Subscribe
ssm:AddTagsToResource
ssm:CreateDocument
ssm:CreateOpsMetadata
ssm:DeleteDocument
ssm:DeleteOpsMetadata
ssm:ModifyDocumentPermission
ssm:RemoveTagsFromResource
ssm:StartAssociationsOnce
ssm:StartAutomationExecution
ssm:UpdateDocument
ssm:UpdateDocumentDefaultVersion
ssm:UpdateOpsMetadata
ssm:UpdateOpsItem
ssm:UpdateServiceSetting
tag:TagResources
tag:UntagResources
```

Pour de plus amples informations sur la création de politiques IAM, consultez Création de politiques IAM dans le guide de l'utilisateur IAM. Pour de plus amples informations sur l'affectation de cette politique à une entité IAM (qui peut être un utilisateur, un groupe ou un rôle), consultez la section Ajout et suppression d'autorisations basées sur l'identité IAM.

Ajout d'applications et de clusters à Application Manager

Application Manager est un composant de AWS Systems Manager. Dans Application Manager, une application est un groupe logique de ressources AWS que vous voulez exploiter en tant qu'unité. Par exemple, ce groupe logique peut représenter différentes versions d'une application, ou les limites de propriété d'opérateurs ou d'environnements de développement.

Lorsque vous sélectionnez Mise en route sur la page d'accueil d'Application Manager, Application Manager importe automatiquement les métadonnées de vos ressources qui ont été créées dans d'autres Services AWS ou fonctionnalités de Systems Manager. Concernant les applications, Application Manager importe des métadonnées de l'ensemble de vos ressources AWS, organisées en groupes de ressources. Chaque groupe de ressources est répertorié dans la catégorie Applications personnalisées en tant qu'application unique. En outre, Application Manager importe automatiquement les métadonnées des ressources qui ont été créées par AWS CloudFormation, AWS Launch Wizard, Amazon Elastic Container Service (Amazon ECS) et Amazon Elastic Kubernetes Service (Amazon EKS). Application Manager affiche ensuite ces ressources dans des catégories prédéfinies.

La liste des Applications inclut les éléments suivants :

- Applications personnalisées
- Launch Wizard
- Stacks CloudFormation
- Applications AppRegistry

La liste des Clusters de conteneurs inclut les éléments suivants :

- Clusters Amazon ECS
- Clusters Amazon EKS

Une fois l'importation terminée, vous pouvez afficher les informations opérationnelles d'une application ou d'une ressource spécifique dans ces catégories prédéfinies. Autrement, si vous voulez ajouter un contexte à propos d'un ensemble de ressources, vous pouvez créer manuellement une application dans Application Manager. Vous pouvez ensuite ajouter des ressources ou des groupes de ressources à cette application. Après avoir créé une application dans Application Manager, vous pouvez afficher les informations opérationnelles relatives à votre ressource dans le contexte d'une application.

Création d'une application dans Application Manager

Procédez comme suit pour créer une application dans Application Manager et ajouter des ressources à cette application.

Pour créer une application dans Application Manager

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Application Manager.
3. Sélectionnez l'onglet Applications, puis Create application (Créer une application).
4. Pour Application name (Nom de l'application), saisissez un nom qui vous aidera à comprendre pourquoi des ressources seront ajoutées à cette application.
5. Pour Application description (Description de l'application), saisissez les informations concernant cette application.
6. Dans la section Choose application components (Choisir des composants d'application), utilisez les options fournies pour choisir des ressources pour cette application. Vous pouvez ajouter une combinaison de ressources, groupes de ressources et piles identifiés, à une application. Vous devez choisir deux composants minimum et 15 maximum. Si vous choisissez des ressources en utilisant des balises, toutes les ressources affectées à ces balises seront répertoriées sous l'onglet Ressources après que la nouvelle application aura été ajoutée. Cela est également vrai pour les ressources incluses dans un groupe de ressources ou dans une pile.

   Si vous ne voyez pas les ressources que vous voulez ajouter à l'application, vérifiez qu'elles ont été balisées correctement, et ajoutées à un groupe AWS Resource Groups ou à une pile AWS CloudFormation.
7. Pour Application tags (Balises d'application) (facultatif), spécifiez les balises pour cette application.
8. Sélectionnez Create (Créer).

Application Manager crée et ouvre l'application. Dans l'arborescence Components (Composants), la nouvelle application est répertoriée comme composant de niveau supérieur, tandis que les ressources, groupes ou piles que vous avez sélectionnés sont répertoriés comme sous-composants. À la prochaine ouverture d'Application Manager, la nouvelle application figurera dans la catégorie Applications personnalisées.

Utilisation des Application Manager

Application Manager est un composant de AWS Systems Manager. Utilisez les rubriques de cette section pour apprendre à travailler avec les applications et clusters Application Manager et à afficher les informations opérationnelles relatives à vos ressources AWS.

Table des matières

Utilisation d'applications

Application Manager est un composant de AWS Systems Manager. Utilisez les rubriques de cette section pour apprendre à travailler avec les applications Application Manager et à afficher les informations opérationnelles relatives à vos ressources AWS.

Table des matières

Affichage des informations de présentation d'une application

Dans Application Manager, un composant de AWS Systems Manager, l'onglet Overview (Présentation) affiche un résumé des alarmes Amazon CloudWatch, des éléments opérationnels (OpsItems), de CloudWatch Application Insights et de l'historique de runbook. Sélectionnez View all (Afficher tout) pour n'importe quelle carte, afin d'ouvrir l'onglet correspondant sous lequel figurent toutes les informations sur l'application, les alarmes, OpsItems, ou l'historique de runbook.

À propos d'Application Insights

CloudWatch Application Insights identifie et paramètre des métriques, des journaux et des alarmes clés sur vos ressources d'application et votre pile technologique. Application Insights surveille en permanence les métriques et les journaux afin de détecter et de corréler les anomalies et les erreurs. Lorsque des erreurs et des anomalies sont détectées, Application Insights génère des CloudWatch Events que vous pouvez utiliser pour paramétrer des notifications ou effectuer des actions. Si vous sélectionnez le bouton Edit configuration (Modifier la configuration) sous l'onglet Monitoring (Surveillance), le système ouvre la console CloudWatch Application Insights. Pour de plus amples informations sur Application Insights, veuillez consulter Qu'est-ce qu'Amazon CloudWatch Application Insights dans le Guide de l'utilisateur Amazon CloudWatch.

A propos de Cost Explorer

Application Manager est intégré à AWS Cost Explorer, une fonction d'AWS Cost Management, via le widget Coût et l'onglet Coût. Une fois que vous avez activé l'explorateur de coûts dans la console Cost Management, le widget Coût et l'onglet Coût dans Application Manager affiche les données de coût pour une application ou un composant d'application non-conteneurisés spécifiques. Vous pouvez appliquer des filtres dans le widget ou l'onglet pour afficher les données de coût selon différentes périodes, différents niveaux de granularité et différents types sous forme de graphique à barres ou linéaire.

Vous pouvez activer cette fonctionnalité en sélectionnant le bouton Accéder à la console AWS Cost Management. Par défaut, les données sont filtrées sur les trois derniers mois. Pour une application non conteneurisée, si vous cliquez sur le bouton Afficher tout de cette section, Application Manager ouvre l'onglet Ressources. Pour les applications conteneurisées, le bouton View all (Afficher tout) ouvre la console AWS Cost Explorer.

Actions que vous pouvez effectuer sur cette page

Vous pouvez activer les widgets suivants et accéder aux informations les concernant dans l'onglet Overview (Présentation) de cette page. Lorsqu'un widget est activé, sélectionnez View all (Afficher tout) pour voir les détails de l'application pertinents pour cette zone.

- Dans la section Insights and Alarms (Informations et alertes), sélectionnez un numéro de sévérité pour ouvrir l'onglet Monitoring (Surveillance) et afficher des détails supplémentaires sur les alertes de la sévérité choisie.

- Dans la section Costs (Coûts), choisissez View all (Afficher tout) pour ouvrir l'onglet Resources (Ressources), dans lequel vous pouvez consulter les données de coût d'une application ou d'un composant d'application spécifique.
- Dans la section Compliance (Conformité), choisissez View all (Afficher tout) pour ouvrir l'onglet Compliance (Conformité), dans lequel vous pouvez consulter les informations de conformité des associations AWS Config et State Manager.

> **Note**
>
> Pour consulter les détails de conformité des correctifs, cliquez directement sur l'onglet Compliance (Conformité). Vous pouvez ensuite consulter les détails de conformité des correctifs pour les nœuds gérés utilisés par l'application sélectionnée.

- Dans la section Runbooks, sélectionnez un runbook pour l'ouvrir sur la page Documents de Systems Manager et afficher des détails supplémentaires sur le document.
- Dans la section OpsItems, sélectionnez une sévérité pour ouvrir l'onglet OpsItems et afficher tous les OpsItems de la sévérité choisie.
- Sélectionnez un bouton View all (Afficher tout) pour ouvrir l'onglet correspondant. Vous pouvez consulter toutes les alarmes, OpsItems ou entrées de l'historique de runbook pour l'application.

Pour ouvrir l'onglet Overview (Présentation)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Application Manager.
3. Dans la section Applications, sélectionnez une catégorie. Pour ouvrir une application que vous avez créée manuellement dans Application Manager, sélectionnez Applications personnalisées.
4. Sélectionnez l'application dans la liste. Application Manager ouvre l'onglet Overview (Présentation).

Utilisation avec vos instances d'application

Application Manager s'intègre à Amazon Elastic Compute Cloud (Amazon EC2) pour afficher des informations sur vos instances dans le contexte d'une application. Application Manager affiche l'état et le statut de l'instance et l'intégrité d'Amazon EC2 Auto Scaling pour une application sélectionnée

dans un format graphique. L'onglet Instances inclut également un tableau contenant les informations suivantes pour chaque instance de votre application :

- État de l'instance (Pending, Stopping, Running, Stopped [En attente, Arrêt, En cours d'exécution, Arrêtée])
- Statut du ping de SSM Agent
- Statut et nom du dernier runbook Systems Manager Automation traité sur l'instance
- Nombre d'alarmes Amazon CloudWatch Logs par État.
  - `ALARM` – La métrique ou l'expression se trouve à l'extérieur du seuil défini.
  - `OK` – La métrique ou l'expression se trouve dans le seuil défini.
  - `INSUFFICIENT_DATA` – L'alerte vient de commencer, la métrique n'est pas disponible, ou la quantité de données n'est pas suffisante pour permettre à la métrique de déterminer le statut de l'alerte.
- Intégrité du groupe Auto Scaling pour les groupes de scalabilité automatique parent et individuel

Si vous choisissez une instance dans le tableau All instances (Toutes les instances), Application Manager affiche les informations relatives à cette instance dans quatre onglets :

- Details (Détails) : tous les détails de l'instance provenant d'Amazon EC2, y compris l'Amazon Machine Image (AMI), les informations DNS, les informations d'adresse IP, etc.
- Health (Intégrité) : le statut d'intégrité actuel tel que fourni par les vérifications de statut du système EC2 et de l'instance.
- Execution history (Historique des exécutions) : journaux d'exécution pour les runbooks de Systems Manager Automation et les appels d'API traités par l'instance.
- CloudWatch alarmes : nom, état, etc., de toutes les CloudWatch alarmes déclenchées par l'instance.

Actions que vous pouvez effectuer sur cette page

Vous pouvez effectuer les actions suivantes sur cette page :

- Lancez, arrêtez et résiliez des instances.
- Appliquez une Chef recette.
- Associez des instances à un groupe Auto Scaling ou détachez-les de celui-ci.
- Activez les mises à jour automatisées pour SSM Agent.

Ouvrir l'onglet Instances

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Application Manager.
3. Dans la section Applications, sélectionnez une catégorie. Si vous souhaitez ouvrir une application que vous avez créée manuellement dans Application Manager, sélectionnez Custom applications (Applications personnalisées).
4. Sélectionnez l'application dans la liste. Application Manager ouvre l'onglet Overview (Présentation).
5. Choisissez l'onglet Instances.

Pour afficher des détails de vos instances d'application

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Application Manager.
3. Dans la section Applications, sélectionnez une catégorie. Si vous souhaitez ouvrir une application que vous avez créée manuellement dans Application Manager, sélectionnez Custom applications (Applications personnalisées).
4. Sélectionnez l'application dans la liste. Application Manager ouvre l'onglet Overview (Présentation).
5. Choisissez l'onglet Instances.
6. Sélectionnez le bouton en regard de l'instance dont vous souhaitez consulter les détails.
7. Vérifiez les détails de l'instance au bas de la page.

Pour mettre à jour automatiquement l'SSM Agent

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Application Manager.
3. Dans la section Applications, sélectionnez une catégorie. Si vous souhaitez ouvrir une application que vous avez créée manuellement dans Application Manager, sélectionnez Custom applications (Applications personnalisées).

4. Sélectionnez l'application dans la liste. Application Manager ouvre l'onglet Overview (Présentation).
5. Choisissez l'onglet Instances.
6. Dans le menu déroulant Actions de l’agent, sélectionnez Configurer la mise à jour de SSM Agent.
7. Choisissez Toutes les instances afin de configurer les mises à jour automatiques de SSM Agent pour toutes les instances gérées. Vous pouvez également choisir Instance afin de configurer les mises à jour automatisées de SSM Agent pour une seule instance de votre application.
8. Sélectionnez le bouton Activer la mise à jour automatique.
9. Dans le menu déroulant Spécifier la planification, choisissez la planification que vous souhaitez utiliser pour les mises à jour de SSM Agent.
10. Sélectionnez Configure (Configurer).

Affichage des ressources d'application

Dans Application Manager, un composant de AWS Systems Manager, l'onglet Resources (Ressources) affiche les ressources AWS dans votre application. Si vous sélectionnez un composant de niveau supérieur, cette page affiche toutes les ressources pour ce composant et tous les sous-composants. Si vous sélectionnez un sous-composant, cette page affiche uniquement les ressources affectées à ce sous-composant.

Actions que vous pouvez effectuer sur cette page

Vous pouvez effectuer les actions suivantes sur cette page :

- Sélectionnez un nom de ressource pour afficher les informations la concernant, notamment les détails fournis par la console sur laquelle elle a été créée, les balises, les alarmes Amazon CloudWatch, les détails AWS Config et les informations de journalisation AWS CloudTrail.
- Sélectionnez le bouton d'option à côté du nom d'une ressource. Ensuite, sélectionnez le bouton Resource timeline (Chronologie des ressources) pour ouvrir la console AWS Config et afficher les informations de conformité relatives à une ressource sélectionnée.
- Si vous avez activé AWS Cost Explorer, la section Cost Explorer affiche les données de coût d'une application ou d'un composant d'application non conteneurisé spécifique. Vous pouvez activer cette fonctionnalité en sélectionnant le bouton Accéder à la console AWS Cost Management. Utilisez les filtres de cette section pour afficher les coûts relatifs à votre application.

Pour ouvrir l'onglet Resources (Ressources)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Application Manager.
3. Dans la section Applications, sélectionnez une catégorie. Pour ouvrir une application que vous avez créée manuellement dans Application Manager, sélectionnez Applications personnalisées.
4. Sélectionnez l'application dans la liste. Application Manager ouvre l'onglet Overview (Présentation).
5. Sélectionnez l'onglet Ressources.

Affichage des informations de conformité

Dans Application Manager, une composante de AWS Systems Manager, la page Configurations affiche des informations sur les ressources AWS Config et sur la conformité des règles de configuration. Cette page affiche également les informations de conformité d'association AWS Systems Manager State Manager. Vous pouvez choisir une ressource, une règle ou une association afin d'ouvrir la console correspondante et obtenir davantage d'informations. Cette page affiche les informations de conformité des 90 derniers jours.

Actions que vous pouvez effectuer sur cette page

Vous pouvez effectuer les actions suivantes sur cette page :

- Sélectionnez un nom de ressource pour ouvrir la console AWS Config et afficher les informations de conformité relatives à une ressource sélectionnée.
- Sélectionnez le bouton d'option à côté du nom d'une ressource. Ensuite, sélectionnez le bouton Resource timeline (Chronologie des ressources) pour ouvrir la console AWS Config et afficher les informations de conformité relatives à une ressource sélectionnée.
- Dans la section Config rules compliance (Conformité des règles de config), vous pouvez effectuer les actions suivantes :
  - Sélectionnez un nom de règle pour ouvrir la console AWS Config et afficher des informations sur cette règle.
  - Sélectionnez Add rules (Ajouter des règles) pour ouvrir la console AWS Config et y créer une règle.

  - Sélectionnez le bouton d'option à côté d'un nom de règle, sélectionnez Actions, puis Manage remediation (Gérer la résolution) pour modifier l'action de résolution d'une règle.
  - Cliquez sur le bouton d'option à côté d'un nom de règle, sélectionnez Actions, puis Re-evaluate (Réévaluer) pour que AWS Config exécute une vérification de conformité sur la règle sélectionnée.
- Dans la section Association compliance (Conformité de l'association), vous pouvez effectuer les actions suivantes :
  - Sélectionnez un nom d'association pour ouvrir la page Associations et afficher des informations sur cette association.
  - Sélectionnez Create association (Créer une association) pour ouvrir Systems Manager State Manager et y créer une association.
  - Cliquez sur le bouton d'option à côté d'un nom d'association et sélectionnez Apply association Appliquer l'association) pour démarrer immédiatement toutes les actions spécifiées dans l'association.

Pour ouvrir l'onglet Compliance (Conformité)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Application Manager.
3. Dans la section Applications, sélectionnez une catégorie. Pour ouvrir une application que vous avez créée manuellement dans Application Manager, sélectionnez Applications personnalisées.
4. Sélectionnez l'application dans la liste. Application Manager ouvre l'onglet Overview (Présentation).
5. Choisissez l'onglet Compliance (Conformité).

Affichage des informations de surveillance

DansApplication Manager, un composant de AWS Systems Manager, l'onglet Monitoring affiche Amazon CloudWatch Application Insights et les détails des alarmes relatives aux ressources d'une application.

À propos d'Application Insights

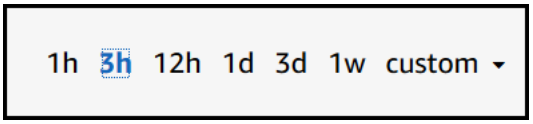
CloudWatch Application Insights identifie et met en place des indicateurs, des journaux et des alarmes clés pour l'ensemble des ressources de votre application et de votre infrastructure

technologique. Application Insights surveille en permanence les métriques et les journaux afin de détecter et de corréler les anomalies et les erreurs. Lorsque le système détecte des erreurs ou des anomalies, Application Insights génère CloudWatch des événements que vous pouvez utiliser pour configurer des notifications ou prendre des mesures. Si vous cliquez sur le bouton Modifier la configuration dans l'onglet Surveillance, le système ouvre la console CloudWatch Application Insights. Pour plus d'informations sur Application Insights, consultez la section Qu'est-ce qu'Amazon CloudWatch Application Insights dans le guide de CloudWatch l'utilisateur Amazon.

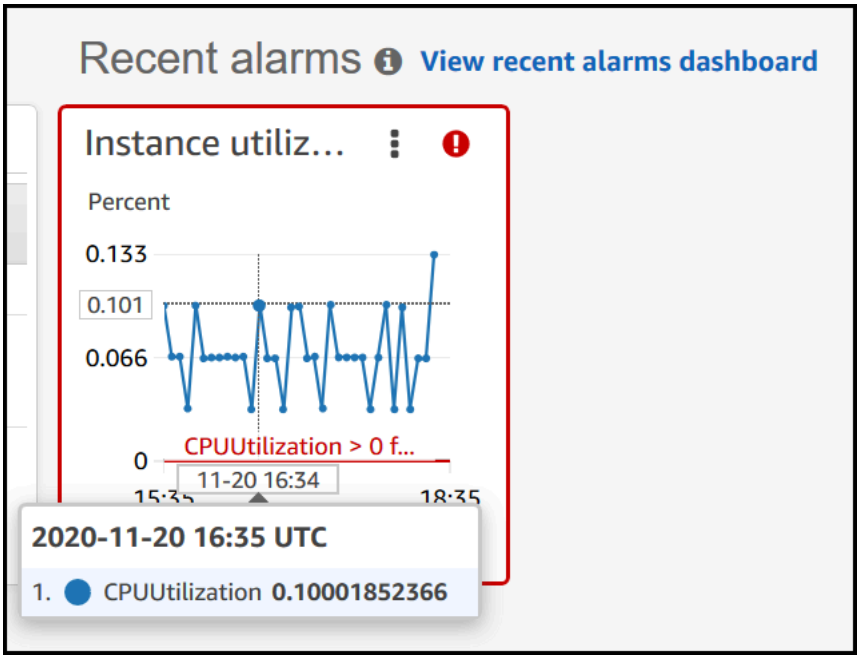
Actions que vous pouvez effectuer sur cette page

Vous pouvez effectuer les actions suivantes sur cette page :

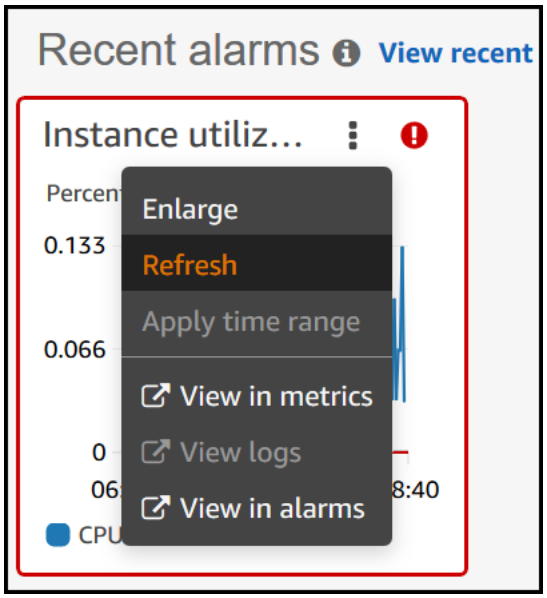
- Choisissez un nom de service dans la section Alarmes par AWS service pour CloudWatch ouvrir le service et l'alarme sélectionnés.
- Ajustez la durée d'affichage des données dans les widgets dans la section Alarmes récentes en sélectionnant l'une des valeurs de durée prédéfinies. Vous pouvez choisir personnalisé pour définir votre propre durée.



- Passez le curseur sur un widget dans la section Alarmes récentes pour afficher une fenêtre contextuelle de données pendant une durée spécifique.

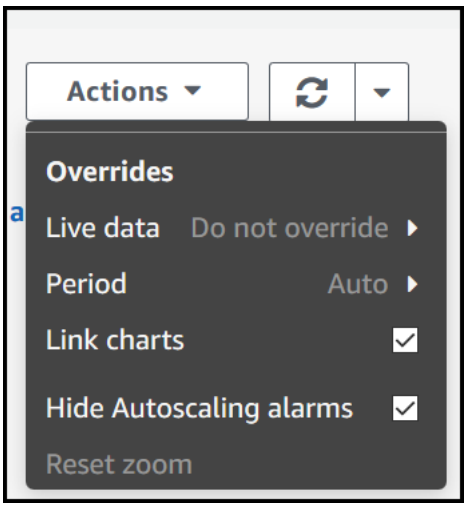


- Sélectionnez le menu d'options d'un widget pour afficher les options d'affichage. Sélectionnez Enlarge (Agrandir) pour développer un widget. Sélectionnez Refresh (Actualiser) pour mettre à jour les données d'un widget. Cliquez et faites glisser le curseur dans un affichage de données de widget pour sélectionner une plage spécifique. Vous pouvez alors choisir Apply time range (Appliquer la plage de temps).



- Sélectionnez le menu Actions pour afficher les options Annuler des données d'alarme, qui incluent les éléments suivants :
  - Sélectionnez si vos widgets de métriques affichent des données en direct. Les données en direct sont des données publiées au cours de la dernière minute et qui n'ont pas été entièrement agrégées. Si les données en direct sont désactivées, seuls les points de données ayant une période d'agrégation d'au moins une minute dans le passé sont affichés. Par exemple, lorsque vous utilisez des périodes de 5 minutes, le point de données de 12h35 est agrégé de 12h35 à 12h40, et affiché à 12h41.

    Si les données en direct sont activées, le point de données le plus récent est affiché dès que les données sont publiées dans l'intervalle d'agrégation correspondant. Chaque fois que vous actualisez l'affichage, le point de données le plus récent peut changer lorsque de nouvelles données de cette période d'agrégation sont publiées.
  - Spécifiez une durée pour les données en direct.
  - Liez les graphiques de la section Alarmes récentes de sorte qu'un zoom avant ou arrière sur un graphique entraîne la même action sur l'autre graphique. Vous pouvez délier les graphiques afin de limiter le zoom à un seul d'entre eux.
  - Masquez les alarmes Auto Scaling.



Pour ouvrir l'onglet Monitoring (Surveillance)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Application Manager.
3. Dans la section Applications, sélectionnez une catégorie. Pour ouvrir une application que vous avez créée manuellement dans Application Manager, sélectionnez Applications personnalisées.
4. Sélectionnez l'application dans la liste. Application Manager ouvre l'onglet Overview (Présentation).
5. Sélectionnez l'onglet Monitoring (Surveillance).

Affichage des OpsItems pour une application

Dans Application Manager, un composant deAWS Systems Manager, l'onglet OpsItems affiche les éléments opérationnels (OpsItems) pour les ressources de l'application sélectionnée. Vous pouvez configurer Systems Manager OpsCenter pour la création automatique d'OpsItems à partir des alarmes Amazon CloudWatch et des événements Amazon EventBridge. Vous pouvez également créer manuellement des éléments OpsItems.

Actions que vous pouvez effectuer dans cet onglet

Vous pouvez effectuer les actions suivantes sur cette page :

- Filtrez la liste d'OpsItems dans le champ de recherche. Vous pouvez filtrer par nom d'OpsItem, ID, ID source ou sévérité. Vous pouvez également filtre la liste en fonction du statut. OpsItems

prend en charge les statuts suivants : Open (Ouvert), In progress (En cours), Open and In progress (Ouvert et En cours), Resolved (Résolu) ou All (Tout).

- Pour modifier le statut d'un OpsItem, sélectionnez le bouton d'option situé à côté de l'élément, puis sélectionnez une option dans le menu Set status (Définir le statut).
- Ouvrez Systems Manager OpsCenter pour créer un OpsItem en choisissant Create (Créer) OpsItem.

Pour ouvrir l'onglet OpsItems

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Application Manager.
3. Dans la section Applications, sélectionnez une catégorie. Pour ouvrir une application que vous avez créée manuellement dans Application Manager, sélectionnez Applications personnalisées.
4. Sélectionnez l'application dans la liste. Application Manager ouvre l'onglet Overview (Présentation).
5. Cliquez sur l'onglet OpsItems.

Affichage des groupes de journaux et des données journalisées

Dans Application Manager, un composant de AWS Systems Manager, l'onglet Journaux affiche une liste des groupes de journaux depuis Amazon CloudWatch Logs.

Actions que vous pouvez effectuer dans cet onglet

Vous pouvez effectuer les actions suivantes sur cette page :

- Sélectionnez un nom de groupe de journaux pour l'ouvrir dans CloudWatch Logs. Vous pouvez ensuite choisir un flux de journaux pour afficher les journaux d'une ressource dans le contexte d'une application.
- Sélectionnez Créer des groupes de journaux pour créer un groupe de journaux dans CloudWatch Logs.

Pour ouvrir l'onglet Logs (Journaux)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Application Manager.
3. Dans la section Applications, sélectionnez une catégorie. Pour ouvrir une application que vous avez créée manuellement dans Application Manager, sélectionnez Applications personnalisées.
4. Sélectionnez l'application dans la liste. Application Manager ouvre l'onglet Overview (Présentation).
5. Sélectionnez l'onglet Logs (Journaux).

Utilisation de runbooks dans Application Manager

Vous pouvez résoudre les problèmes avec les ressources AWS à partir d'Application Manager, une fonctionnalité de AWS Systems Manager, en utilisant des runbooks Automation. Un runbook Automation définit les actions effectuées par Systems Manager sur vos instances gérées et d'autres ressources AWS lors de l'exécution d'une automatisation. Automation est une fonctionnalité de AWS Systems Manager. Un runbook 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.

Lorsque vous choisissez Start runbook (Démarrer un runbook) à partir d'une application ou un cluster Application Manager, le système affiche une liste filtrée de runbooks disponibles en fonction du type de ressources de votre application ou votre cluster. Lorsque vous sélectionnez le runbook à démarrer, Systems Manager ouvre la page Exécuter le document d'automatisation.

Application Manager inclut les améliorations suivantes apportées à l'utilisation de runbooks.

- Si vous sélectionnez le nom d'une ressource dans Application Manager, puis que vous sélectionnez Exécuter un runbook, le système affiche une liste de runbooks filtrée en fonction de ce type de ressource.
- Vous pouvez lancer une automatisation sur toutes les ressources du même type en choisissant un runbook dans la liste, puis en sélectionnant Exécuter pour les ressources de même type.

Avant de commencer

Avant de démarrer un runbook à partir de Application Manager, effectuez la procédure suivante :

- Vérifiez que vous disposez des autorisations voulues pour démarrer des runbooks. Pour de plus amples informations, veuillez consulter Configuration d'Automation.
- Consultez la documentation sur la procédure d'automatisation associée au démarrage des runbooks. Pour de plus amples informations, veuillez consulter Exécution d'automatisations.

Pour démarrer un runbook à partir de Application Manager

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Application Manager.
3. Dans la section Applications, sélectionnez une catégorie. Pour ouvrir une application que vous avez créée manuellement dans Application Manager, sélectionnez Applications personnalisées.
4. Sélectionnez l'application dans la liste. Application Manager ouvre l'onglet Overview (Présentation).
5. Choisissez Démarrer le runbook. Application Manager ouvre la fenêtre contextuelle du Widget d'automatisation. Pour obtenir des informations sur les options du Widget d'automatisation, veuillez consulter la rubrique Exécution d'automatisations.

Utilisation des balises dans Application Manager

Vous pouvez ajouter ou supprimer des balises rapidement sur des applications et des ressources AWS dans Application Manager. Pour en savoir plus sur les identifications, consultez Balisage des ressources Systems Manager.

Procédez comme suit pour ajouter ou supprimer une balise d'une application, ainsi que toutes les ressources AWS de cette application.

Pour ajouter ou supprimer une balise d'une application et de toutes les ressources de l'application

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Application Manager.
3. Dans la section Applications, sélectionnez une catégorie. Pour ouvrir une application que vous avez créée manuellement dans Application Manager, sélectionnez Applications personnalisées.
4. Sélectionnez l'application dans la liste. Application Manager ouvre l'onglet Overview (Présentation).

5. Dans la section Informations d'application, sélectionnez le numéro situé sous Balises d'application. Si aucune balise n'est affectée à l'application, le numéro affiche zéro.
6. Pour ajouter une balise, sélectionnez Add new tag (Ajouter une nouvelle balise). Spécifiez une clé et une valeur facultative. Pour supprimer une balise, sélectionnez Remove (Supprimer).
7. Sélectionnez Enregistrer.

Procédez comme suit pour ajouter ou supprimer une balise d'une ressource spécifique dans Application Manager.

Pour ajouter ou supprimer une balise d'une ressource

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Application Manager.
3. Dans la section Applications, sélectionnez une catégorie. Pour ouvrir une application que vous avez créée manuellement dans Application Manager, sélectionnez Applications personnalisées.
4. Sélectionnez l'application dans la liste. Application Manager ouvre l'onglet Overview (Présentation).
5. Sélectionnez l'onglet Ressources.
6. Sélectionnez un nom de ressource.
7. Dans la section Tags (Balises), sélectionnez Edit (Modifier).
8. Pour ajouter une balise, sélectionnez Add new tag (Ajouter une nouvelle balise). Spécifiez une clé et une valeur facultative. Pour supprimer une balise, sélectionnez Remove (Supprimer).
9. Choisissez Enregistrer.

Utilisation des modèles et des modèles AWS CloudFormation dans Application Manager.

Application Manager, une fonctionnalité de AWS Systems Manager, vous aide à approvisionner et à gérer les ressources pour vos applications via l'intégration à AWS CloudFormation. Vous pouvez créer, modifier et supprimer des modèles et des piles AWS CloudFormation dans Application Manager. Une pile est un ensemble de ressources AWS que vous gérez comme une seule unité. Cela signifie que vous pouvez créer, mettre à jour ou supprimer un ensemble de ressources AWS en utilisant des piles CloudFormation. Un modèle est un fichier texte au format JSON ou YAML, qui spécifie les ressources à approvisionner dans vos piles.

Application Manager inclut également une bibliothèque de modèles dans laquelle vous pouvez cloner, créer et stocker des modèles. Application Manager et CloudFormation affichent les mêmes informations à propos de l'état actuel d'une pile. Les modèles et les mises à jour de modèles sont stockés dans Systems Manager jusqu'à ce que vous approvisionniez la pile. Ensuite, les modifications s'affichent également dans CloudFormation.

Après avoir créé une pile dans Application Manager, la page piles CloudFormation affiche des informations utiles à ce sujet. Cela inclut le modèle utilisé pour créer la pile, un nombre d'OpsItems pour les ressources de votre pile, le statut de la pile, et le statut de l'écart.

Démarrage de Cost Explorer

Application Manager est intégré à AWS Cost Explorer, une fonctionnalité de AWS Cost Management, via le widget Cost (Coût). Une fois que vous avez activé l'explorateur de coûts dans la console Cost Management, le widget Cost (Coût) dans Application Manager affiche les données de coût pour une application ou un composant d'application non-conteneurisés spécifiques. Vous pouvez appliquer des filtres dans le widget pour afficher les données de coût selon différentes périodes, différentes granularités et différents types sous forme de graphique à barres ou linéaire.

Vous pouvez activer cette fonctionnalité en sélectionnant le bouton Accéder à la console AWS Cost Management. Par défaut, les données sont filtrées sur les trois derniers mois. Pour une application non conteneurisée, si vous cliquez sur le bouton Afficher tout de cette section, Application Manager ouvre l'onglet Ressources. Pour les applications conteneurisées, le bouton View all (Afficher tout) ouvre la console AWS Cost Explorer.

> **Note**
>
> Cost Explorer utilise des balises pour le suivi des coûts de vos applications. Si votre application basée sur la pile AWS CloudFormation n'est pas configurée avec la clé de balise `AppManagerCFNStackKey`, il est impossible pour Cost Explorer de présenter des données de coûts précises dans Application Manager. Lorsque la clé de balise `AppManagerCFNStackKey` n'est pas détectée, la console vous invite à ajouter la balise à votre pile CloudFormation afin de permettre le suivi des coûts. Son ajout fait correspondre la clé de balise à l'Amazon Resource Name (ARN) de votre pile et permet au widget Cost (Coût) d'afficher des données de coûts précises.

> **Important**
>
> L'ajout de la balise `AppManagerCFNStackKey` déclenchera une mise à jour de la pile. Les configurations manuelles effectuées après le déploiement initial de la pile ne seront pas reflétées après l'ajout de la balise utilisateur. Pour plus d'informations sur les comportements de mise à jour des ressources, veuillez consulter la rubrique Comportements de mise à jour des ressources de pile dans le Guide de l'utilisateur AWS CloudFormation.

Avant de commencer

Utilisez les liens suivants pour en savoir plus sur les concepts CloudFormation avant de créer, de modifier ou de supprimer des modèles et des piles CloudFormation en utilisant Application Manager.

- Présentation de AWS CloudFormation
- Bonnes pratiques AWS CloudFormation
- Découvrir les concepts de base des modèles
- Utilisation des piles AWS CloudFormation
- Utilisation des modèles AWS CloudFormation
- Exemples de modèle

Rubriques

- Utilisation des modèles CloudFormation
- Utilisation des piles CloudFormation

Utilisation des modèles CloudFormation

Application Manager, une fonctionnalité de AWS Systems Manager, inclut une bibliothèque de modèles et d'autres outils pour vous aider à gérer des modèles AWS CloudFormation. Cette section comprend les informations suivantes.

Rubriques

- Utilisation de la bibliothèque de modèles
- Création d'un modèle
- Modification d'un modèle

Utilisation de la bibliothèque de modèles

La bibliothèque de modèles Application Manager vous fournit des outils pour afficher, créer, modifier, supprimer et cloner des modèles. Vous pouvez également approvisionner des piles directement à partir de la bibliothèque de modèles. Les modèles sont stockés sous forme de documents Systems Manager (SSM) du type `CloudFormation`. En stockant des modèles sous forme de documents SSM, vous pouvez utiliser des contrôles de version pour utiliser différentes versions d'un modèle. Vous pouvez également définir des autorisations et partager des modèles. Une fois l'approvisionnement d'une pile réussi, la pile et le modèle sont disponibles dans Application Manager et CloudFormation.

Avant de commencer

Nous vous recommandons de lire les rubriques suivantes pour en savoir plus sur les documents SSM avant de commencer à travailler avec les modèles CloudFormation dans Application Manager.

- AWS Systems Manager Documents
- Partage de documents SSM
- Bonnes pratiques pour les documents SSM partagés

Pour afficher la bibliothèque de modèles dans Application Manager

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Application Manager.
3. Dans la section Applications, sélectionnez piles CloudFormation.
4. Sélectionnez Bibliothèque de modèles.

Création d'un modèle

La procédure suivante décrit la création d'un modèle CloudFormation dans Application Manager. Lorsque vous créez un modèle, vous saisissez les détails de la pile du modèle au format JSON ou YAML. Si le format JSON ou YAML ne vous est pas familier, vous pouvez utiliser AWS CloudFormation Designer, un outil destiné à la création et la modification visuelles de modèles. Pour plus d'informations, consultez Qu'est-ce que AWS CloudFormation Designer ? dans le AWS CloudFormationGuide de l'utilisateur. Pour plus d'informations sur la structure et la syntaxe d'un modèle, consultez Anatomie du modèle.

Vous pouvez également créer un modèle à partir de plusieurs extraits de modèle. Les extraits de modèle proposent des exemples qui montrent comment écrire des modèles pour une ressource déterminée. Par exemple, vous pouvez afficher des extraits pour les instances Amazon Elastic Compute Cloud (Amazon EC2), les domaines Amazon Simple Storage Service (Amazon S3), les mappages AWS CloudFormation, etc. Les extraits sont regroupés par ressource. Vous pouvez trouver des extraits AWS CloudFormation polyvalents dans la section Extraits de modèle généraux du Guide de l'utilisateur AWS CloudFormation.

Création d'un modèle CloudFormation dans Application Manager (console)

Procédez comme suit pour créer un modèle CloudFormation dans Application Manager en utilisant la AWS Management Console.

Pour créer un modèle CloudFormation dans Application Manager

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Application Manager.
3. Dans la section Applications, sélectionnez piles CloudFormation.
4. Sélectionnez Bibliothèque de modèles, puis Créer un modèle, ou bien sélectionnez un modèle existant, puis Actions, Clone.
5. Pour Nom, saisissez un nom pour le modèle, qui vous aidera à identifier les ressources créées ou le but de la pile.
6. (Facultatif) Pour Nom de la version, saisissez un nom ou un numéro pour identifier la version du modèle.
7. (Facultatif) Pour Description, saisissez les informations concernant ce modèle.
8. Dans la section Éditeur de code, sélectionnez YAML ou JSON, puis saisissez ou copiez et collez votre code de modèle.
9. (Facultatif) Dans la section Tags (Balises), appliquez une ou plusieurs paires nom/valeur de clé de balise au modèle.

   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 plus d'informations sur le balisage des ressources Systems Manager, consultez Balisage des ressources Systems Manager.

10. (Facultatif) Dans la section Autorisations, saisissez un ID Compte AWS et sélectionnez Ajouter un compte. Cette action fournit une autorisation en lecture au modèle. Le propriétaire du compte peut approvisionner et cloner le modèle, mais il ne peut ni le modifier ni le supprimer.
11. Sélectionnez Create (Créer). Le modèle est enregistré dans le service Document Systems Manager (SSM).

Création d'un modèle CloudFormation dans Application Manager (ligne de commande)

Après avoir créé le contenu de votre modèle CloudFormation au format JSON ou YAML, vous pouvez utiliser la AWS Command Line Interface (AWS CLI) ou AWS Tools for PowerShell pour enregistrer le modèle sous forme de document SSM. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

Avant de commencer

Installez et configurez l'AWS CLI ou AWS Tools for PowerShell si vous ne l'avez pas déjà fait. Pour plus d'informations, consultez la section Installation ou mise à jour de la version la plus récente de l'AWS CLI et Installation d'AWS Tools for PowerShell.

Linux & macOS

```
aws ssm create-document \
    --content file://path/to/template_in_json_or_yaml \
    --name "a_name_for_the_template" \
    --document-type "CloudFormation" \
    --document-format "JSON_or_YAML" \
    --tags "Key=tag-key,Value=tag-value"
```

Windows

```
aws ssm create-document ^
    --content file://C:\path\to\template_in_json_or_yaml ^
    --name "a_name_for_the_template" ^
    --document-type "CloudFormation" ^
    --document-format "JSON_or_YAML" ^
    --tags "Key=tag-key,Value=tag-value"
```

PowerShell

```
$json = Get-Content -Path "C:\path\to\template_in_json_or_yaml | Out-String
```

```
New-SSMDocument `
    -Content $json `
    -Name "a_name_for_the_template" `
    -DocumentType "CloudFormation" `
    -DocumentFormat "JSON_or_YAML" `
    -Tags "Key=tag-key,Value=tag-value"
```

Si elle aboutit, la commande renvoie une réponse semblable à la suivante :

```
{
    "DocumentDescription": {
        "Hash": "c1d9640f15fbdba6deb41af6471d6ace0acc22f213bdd1449f03980358c2d4fb",
        "HashType": "Sha256",
        "Name": "MyTestCFTemplate",
        "Owner": "428427166869",
        "CreatedDate": "2021-06-04T09:44:18.931000-07:00",
        "Status": "Creating",
        "DocumentVersion": "1",
        "Description": "My test template",
        "PlatformTypes": [],
        "DocumentType": "CloudFormation",
        "SchemaVersion": "1.0",
        "LatestVersion": "1",
        "DefaultVersion": "1",
        "DocumentFormat": "YAML",
        "Tags": [
            {
                "Key": "Templates",
                "Value": "Test"
            }
        ]
    }
```

Modification d'un modèle

Procédez comme suit pour modifier un modèle CloudFormation dans Application Manager. Les modifications de modèle sont disponibles dans CloudFormation après l'approvisionnement d'une pile qui utilise le modèle mis à jour.

Pour modifier un modèle CloudFormation dans Application Manager

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Application Manager.
3. Dans la section Applications, sélectionnez piles CloudFormation.
4. Sélectionnez Bibliothèque de modèles.
5. Sélectionnez une règle, puis sélectionnez Actions, Edit (Modifier). Vous ne pouvez pas modifier le nom d'un modèle, mais tous les autres détails sont modifiables.
6. Sélectionnez Enregistrer. Le modèle est enregistré dans le service Document Systems Manager.

Utilisation des piles CloudFormation

Application Manager, une fonctionnalité de AWS Systems Manager, vous aide à approvisionner et à gérer les ressources pour vos applications via l'intégration à AWS CloudFormation. Vous pouvez créer, modifier et supprimer des modèles et des piles CloudFormation dans Application Manager. Une pile est un ensemble de ressources AWS que vous gérez comme une seule unité. Cela signifie que vous pouvez créer, mettre à jour ou supprimer un ensemble de ressources AWS en utilisant des piles CloudFormation. Un modèle est un fichier texte au format JSON ou YAML, qui spécifie les ressources à approvisionner dans vos piles. Cette section comprend les informations suivantes.

Rubriques

Création d'une pile

La procédure suivante décrit la création d'une pile CloudFormation avec Application Manager. Une pile est basée sur un modèle. Lorsque vous créez une pile, vous pouvez choisir un modèle existant ou en créer un. Après avoir créé la pile, le système tente immédiatement de créer les ressources identifiées dans la pile. Après que le système a approvisionné correctement les ressources, le modèle et la pile peuvent être affichés et modifiés dans Application Manager et CloudFormation.

> **Note**
>
> L'utilisation d'Application Manager pour créer une pile n'induit aucuns frais, mais les ressources AWS créées dans la pile vous seront facturées.

Création d'une pile CloudFormation en utilisant Application Manager (console)

Utilisez la procédure suivante pour créer une pile à l'aide du Application Manager dans la AWS Management Console.

Pour créer une pile CloudFormation

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Application Manager.
3. Dans la section Applications, sélectionnez piles CloudFormation.
4. Dans la section Préparer un modèle, sélectionnez une option. Si vous sélectionnez Utiliser un modèle existant, les onglets de la section Choisir un modèle vous aideront à localiser le modèle à utiliser. Si vous sélectionnez l'une des autres options, suivez l'assistant pour préparer un modèle.
5. Sur la page Spécifier les détails du modèle, vérifiez les détails du modèle pour vous assurer que le processus crée les ressources voulues.
   - (Facultatif) Dans la section Tags (Balises), appliquez une ou plusieurs paires nom/valeur de clé de balise au modèle.
   - 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 plus d'informations sur le balisage des ressources Systems Manager, consultez Balisage des ressources Systems Manager.
   - Sélectionnez Suivant.
6. Sur la page Modifier les détails de la pile, pour Nom de la pile, saisissez un nom qui vous aidera à identifier les ressources créées par la pile ou son but.
   - La section Paramètres inclut tous les paramètres facultatifs et obligatoires spécifiés dans le modèle. Saisissez un ou plusieurs paramètres dans chaque champ.

- (Facultatif) Dans la section Tags (Balises), appliquez une ou plusieurs paires nom/valeur de clé de balise à la pile.
- (Facultatif) Dans la section Autorisations, spécifiez un nom de rôle AWS Identity and Access Management (IAM) ou un Amazon Resource Name (ARN) IAM. Le système utilise le rôle de service spécifié pour créer l'ensemble des ressources spécifiées dans votre pile. Si vous ne définissez aucun rôle IAM, AWS CloudFormation utilise alors une session temporaire générée par le système à partir de vos informations d'identification utilisateur. Pour de plus amples informations sur ce rôle IAM, consultez Rôle de service AWS CloudFormation dans le Guide de l'utilisateur AWS CloudFormation.
- Sélectionnez Suivant.

7. Sur la page Examen et approvisionnement, examinez les détails de la pile. Sélectionnez un bouton Modifier sur cette page pour apporter des modifications.
8. Sélectionnez Approvisionner une pile.

Application Manager affiche la page Piles CloudFormation, ainsi que le statut de création et de déploiement de la pile. Si la création et l'approvisionnement de la pile par CloudFormation échouent, consultez les rubriques suivantes dans le Guide de l'utilisateur AWS CloudFormation.

- Codes d'état de la pile
- Résolutions des problèmes liés à AWS CloudFormation

Une fois que vos ressources de pile sont approvisionnées et en cours d'exécution, les utilisateurs peuvent les modifier directement en utilisant le service sous-jacent qui a créé la ressource. Par exemple, un utilisateur peut utiliser la console Amazon Elastic Compute Cloud (Amazon EC2) pour mettre à jour une instance de serveur qui a été créée dans le cadre d'une pile CloudFormation. Certaines modifications peuvent être accidentelles, et d'autres peuvent être apportées intentionnellement pour répondre aux événements opérationnels prioritaires. Quoi qu'il en soit, les modifications apportées en dehors de CloudFormation peuvent compliquer les opérations de mise à jour et de suppression de pile. Vous pouvez utiliser la détection de l'écart ou le statut de l'écart pour identifier les ressources d'une pile pour lesquelles des modifications de configuration ont été apportées en dehors de la gestion CloudFormation. Pour de plus amples informations sur le statut de l'écart, consultez Détection de modifications non gérées de la configuration des piles et des ressources.

Création d'une pile CloudFormation en utilisant la Application Manager (ligne de commande)

Suivez la procédure AWS Command Line Interface (AWS CLI) pour approvisionner une pile en utilisant un modèle CloudFormation stocké sous forme de document SSM dans Systems Manager. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations. Pour obtenir des informations sur d'autres procédures AWS CLIassociées à la création de piles, consultez Création d'une pile dans le Guide de l'utilisateur AWS CloudFormation.

Avant de commencer

Installez et configurez l'AWS CLI ou AWS Tools for PowerShell si vous ne l'avez pas déjà fait. Pour plus d'informations, consultez la section Installation ou mise à jour de la version la plus récente de l'AWS CLI et Installation d'AWS Tools for PowerShell.

Linux & macOS

```
aws cloudformation create-stack \
    --stack-name a_name_for_the_stack \
    --template-url "ssm-doc://arn:aws:ssm:Region:account_ID:document/template_name"
 \
```

Windows

```
aws cloudformation create-stack ^
     --stack-name a_name_for_the_stack ^
     --template-url "ssm-doc://arn:aws:ssm:Region:account_ID:document/template_name"
 ^
```

PowerShell

```
New-CFNStack `
    -StackName "a_name_for_the_stack" `
    -TemplateURL "ssm-doc://arn:aws:ssm:Region:account_ID:document/template_name" `
```

Mise à jour d'une pile

Vous pouvez déployer des mises à jour sur une pile CloudFormation en modifiant directement la pile dans Application Manager. Avec une mise à jour directe, vous spécifiez les mises à jour à apporter à un modèle ou à des paramètres d'entrée. Après avoir enregistré et déployé les modifications,

CloudFormation met à jour les ressources AWS en fonction des modifications que vous avez spécifiées.

Vous pouvez visualiser les modifications que CloudFormation apportera à la pile avant d'effectuer la mise à jour à l'aide des jeux de modifications. Pour de plus amples informations, consultez Mise à jour des piles à l'aide de jeux de modifications dans le Guide de l'utilisateur AWS CloudFormation.

Pour mettre à jour une pile CloudFormation dans Application Manager

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Application Manager.
3. Dans la section Applications, sélectionnez piles CloudFormation.
4. Sélectionnez une pile dans la liste, puis sélectionnez Actions, Mettre à jour la pile.
5. Sur la page Spécifier la source du modèle, sélectionnez l'une des options suivantes, puis sélectionnez Suivant.
   - Sélectionnez Utiliser le code de modèle actuellement approvisionné dans la pile pour afficher un modèle. Sélectionnez une version de modèle dans la liste Versions, puis sélectionnez Next (Suivant).
   - Sélectionnez Basculer vers un autre modèle pour choisir un modèle ou en créer un pour la pile.
6. Une fois les modifications apportées, sélectionnez Suivant.
7. Sur la page Modifier les détails de la pile, vous pouvez modifier les paramètres, les balises et les autorisations. Vous ne pouvez pas changer le nom de la pile. Effectuez les modifications de votre choix, puis sélectionnez Next (Suivant).
8. Sur la page Examen et approvisionnement, examinez les détails de la pile, puis sélectionnez Approvisionner une pile.

Utilisation des clusters dans Application Manager

Utilisez les rubriques de cette section pour apprendre à travailler avec les clusters de conteneurs Amazon Elastic Container Service (Amazon ECS) et Amazon Elastic Kubernetes Service (Amazon EKS) dans Application Manager, un composant de AWS Systems Manager.

Table des matières

- Utiliser Amazon ECS dans Application Manager
- Utiliser Amazon EKS dans Application Manager
- Utiliser des runbooks pour des clusters

Utiliser Amazon ECS dans Application Manager

Grâce à Application Manager cette fonctionnalité AWS Systems Manager, vous pouvez visualiser et gérer votre infrastructure de cluster Amazon Elastic Container Service (Amazon ECS). Application Managerapplique une balise à votre cluster Amazon ECS en utilisant le nom de ressource Amazon (ARN) du cluster comme valeur de balise. Application Managerfournit une vue d'exécution des composants des ressources de calcul, de mise en réseau et de stockage d'un cluster.

> **Note**
>
> Il ne vous est pas possible de gérer ou d'afficher les informations opérationnelles relatives à vos conteneurs dans Application Manager. Vous ne pouvez gérer et afficher que les informations opérationnelles relatives à l'infrastructure qui héberge vos ressources Amazon ECS.

Actions que vous pouvez effectuer sur cette page

Vous pouvez effectuer les actions suivantes sur cette page :

- Sélectionnez Gérer un cluster pour ouvrir le cluster dans Amazon ECS.
- Sélectionnez Afficher tout pour afficher la liste des ressources de votre cluster.
- Choisissez Afficher dans CloudWatch pour afficher les alarmes relatives aux ressources sur Amazon CloudWatch.
- Choisissez Manage nodes (Gérer des nœuds) ou Manage Fargate profiles (Gérer les profils Fargate) pour afficher ces ressources dans Amazon ECS.
- Sélectionnez un ID de ressource pour afficher des informations détaillées à ce sujet dans la console où il a été créé.
- Affichez une liste d'OpsItems associés à vos clusters.
- Affichez un historique des runbooks qui ont été exécutés sur vos clusters.

Pour ouvrir un cluster ECS

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Application Manager.
3. Dans la section Clusters de conteneurs, sélectionnez Cluster ECS.
4. Sélectionnez un cluster dans la liste. Application Manager ouvre l'onglet Overview (Présentation).

Utiliser Amazon EKS dans Application Manager

Application Manager, une fonctionnalité de AWS Systems Manager, s'intègre à Amazon Elastic Kubernetes Service (Amazon EKS) pour fournir des informations sur l'état de votre infrastructure de cluster Amazon EKS. Application Managerapplique une balise à votre cluster Amazon EKS en utilisant le nom de ressource Amazon (ARN) du cluster comme valeur de balise. Application Managerfournit une vue d'exécution des composants des ressources de calcul, de réseau et de stockage d'un cluster.

> **Note**
>
> Il ne vous est pas possible de gérer ou d'afficher les informations opérationnelles relatives à vos pods ou conteneurs Amazon EKS dans Application Manager. Vous ne pouvez gérer et afficher que les informations opérationnelles relatives à l'infrastructure qui héberge vos ressources Amazon EKS.

Actions que vous pouvez effectuer sur cette page

Vous pouvez effectuer les actions suivantes sur cette page :

- Sélectionnez Manage cluster (Gérer un cluster) pour ouvrir le cluster dans Amazon EKS.
- Sélectionnez Afficher tout pour afficher la liste des ressources de votre cluster.
- Choisissez Afficher dans CloudWatch pour afficher les alarmes relatives aux ressources sur Amazon CloudWatch.
- Sélectionnez Manage nodes (Gérer des nœuds) ou Manage Fargate profiles (Gérer les profils Fargate) pour afficher ces ressources dans Amazon EKS.

- Sélectionnez un ID de ressource pour afficher des informations détaillées à ce sujet dans la console où il a été créé.
- Affichez une liste d'OpsItems associés à vos clusters.
- Affichez un historique des runbooks qui ont été exécutés sur vos clusters.

Pour ouvrir une application clusters EKS

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Application Manager.
3. Dans la section Clusters de conteneurs, sélectionnez Cluster EKS.
4. Sélectionnez un cluster dans la liste. Application Manager ouvre l'onglet Overview (Présentation).

Utiliser des runbooks pour des clusters

Vous pouvez résoudre les problèmes avec les ressources AWS à partir d'Application Manager, une fonctionnalité de AWS Systems Manager, en utilisant des runbooks Systems Manager Automation. Lorsque vous sélectionnez Démarrer un runbook à partir d'n cluster Application Manager, le système affiche une liste de runbooks filtrée en fonction du type de ressources de votre cluster. Lorsque vous sélectionnez le runbook à démarrer, Systems Manager ouvre la page Exécuter le document d'automatisation.

Avant de commencer

Avant de démarrer un runbook à partir de Application Manager, effectuez la procédure suivante :

- Vérifiez que vous disposez des autorisations voulues pour démarrer des runbooks. Pour de plus amples informations, veuillez consulter Configuration d'Automation.
- Consultez la documentation sur la procédure d'automatisation associée au démarrage des runbooks. Pour de plus amples informations, veuillez consulter Exécution d'automatisations.
- Si vous envisagez de lancer des runbooks sur plusieurs ressources à la fois, consultez la documentation sur l'utilisation des cibles et des contrôles de débit. Pour de plus amples informations, veuillez consulter Exécution des automatisations à grande échelle.

Pour démarrer un runbook pour des clusters à partir d'Application Manager

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Application Manager.
3. Dans la section Clusters de conteneurs, sélectionnez un type de conteneur.
4. Sélectionnez le cluster dans la liste. Application Manager ouvre l'onglet Overview (Présentation).
5. Dans l'onglet Runbooks, sélectionnez Start runbook (Démarrer un runbook). Application Manager ouvre la page Execute automation document (Exécuter le document d'automatisation) sous un nouvel onglet. Pour obtenir des informations sur les options de la page Exécuter le document d'automatisation, veuillez consulter Exécution d'automatisations.

AWS AppConfig

AWS AppConfig les indicateurs de fonctionnalités et les configurations dynamiques aident les concepteurs de logiciels à ajuster rapidement et en toute sécurité le comportement des applications dans les environnements de production sans déploiement de code complet. AWS AppConfig accélère la fréquence de publication des logiciels, améliore la résilience des applications et vous aide à résoudre les problèmes émergents plus rapidement. Grâce aux indicateurs de fonctionnalités, vous pouvez progressivement proposer de nouvelles fonctionnalités aux utilisateurs et mesurer l'impact de ces modifications avant de déployer complètement les nouvelles fonctionnalités pour tous les utilisateurs. Grâce aux indicateurs opérationnels et aux configurations dynamiques, vous pouvez mettre à jour les listes de blocage, les listes d'autorisation, les limites de limitation, la verbosité de journalisation et effectuer d'autres réglages opérationnels pour répondre rapidement aux problèmes dans les environnements de production.

Pour plus d'informations, voir Qu'est-ce que c'est AWS AppConfig ? dans le guide de AWS AppConfig l'utilisateur.

AWS Systems Manager Parameter Store

Parameter Store, une fonctionnalité de AWS Systems Manager, fournit un stockage hiérarchique sécurisé pour la gestion des données de configuration et la gestion des secrets. Vous pouvez stocker des données telles que des mots de passe, des chaînes de base de données, des ID d'Amazon Machine Image (AMI) et des codes de licence en tant que valeurs de paramètres. 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. Pour vos premiers pas dans Parameter Store, ouvrez Systems Manager console. Dans le panneau de navigation, sélectionnez Parameter Store.

Parameter Store est également intégré à Secrets Manager. Vous pouvez récupérer les secrets Secrets Manager lors de l'utilisation d'autres Services AWS qui prennent déjà en charge les références aux paramètres Parameter Store. Pour plus d'informations, consultez Référencement des secrets AWS Secrets Manager à partir des paramètres Parameter Store.

> **Note**
>
> Pour implémenter les cycles de vie de rotation des mots de passe, utilisez. AWS Secrets Manager Vous pouvez effectuer une rotation, gérer et récupérer les informations d'identification de la base de données, les clés d'API et d'autres secrets tout au long de leur cycle de vie à l'aide de Secrets Manager. Pour plus d'informations, voir Qu'est-ce que c'est AWS Secrets Manager ? dans le guide de AWS Secrets Manager l'utilisateur.

Comment mon organisation peut-elle tirer parti de Parameter Store ?

Parameter Store offre les avantages suivants :

- Utilisation d'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.
- Stockez les paramètres de manière fiable car Parameter Store est hébergé dans plusieurs zones de disponibilité dans une Région AWS.

À qui est destiné Parameter Store ?

- Tout AWS client qui souhaite disposer d'une méthode centralisée pour gérer les données de configuration.

- Les développeurs de logiciels qui veulent stocker différentes connexions et différents flux de référence.
- Les administrateurs qui veulent recevoir des notifications lorsque leurs secrets et leurs mots de passe sont ou ne sont pas modifiés.

Quelles sont les fonctions d'Parameter Store ?

- Notification de modification

  Vous pouvez configurer des notifications de modifications et invoquer des actions automatisées à la fois pour les paramètres et les politiques de paramètres. Pour plus d’informations, consultez Configuration de notifications ou d'actions de déclenchement basées sur des événements Parameter Store.

- Organisation des paramètres

  Vous pouvez baliser vos paramètres individuellement pour vous aider à identifier un ou plusieurs paramètres en fonction des balises que vous leur avez affectées. Par exemple, vous pouvez baliser des paramètres pour des environnements ou des services spécifiques. Pour plus d’informations, consultez Balisage de paramètres Systems Manager.

- Versions d'étiquettes

  Vous pouvez associer un alias aux 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 désignant une instance Amazon Elastic Compute Cloud (Amazon EC2), et Parameter Store les valide pour s'assurer qu'il fait référence au type de ressource attendu, que la ressource existe et que le client a l'autorisation d'utiliser la ressource. Par exemple, vous pouvez créer un paramètre avec un ID d'Amazon Machine Image (AMI) comme valeur avec le type de données `aws:ec2:image`, et Parameter Store 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 la valeur spécifiée AMI est disponible dans votre Compte AWS.

- Secrets de référence

Parameter Storeest intégré AWS Secrets Manager afin que vous puissiez récupérer les secrets de Secrets Manager lorsque vous en utilisez un autre Services AWS qui prend déjà en charge les références aux Parameter Store paramètres.

- Partage de paramètres avec d'autres comptes

  Vous pouvez éventuellement centraliser les données de configuration en un seul Compte AWS et partager les paramètres avec d'autres comptes qui ont besoin d'y accéder.

- Accessible depuis d'autres Services AWS

  Vous pouvez utiliser les paramètres Parameter Store avec d'autres fonctionnalités de Systems Manager et Services AWS pour récupérer les secrets et les données de configuration à partir d'un magasin central. Les paramètres fonctionnent avec les fonctionnalités de Systems Manager telles que Run Command l'automatisation et State Manager les fonctionnalités de AWS Systems Manager. Vous pouvez également référencer des paramètres dans plusieurs autres domaines 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

- Intégrez avec d'autres Services AWS

  Configurez l'intégration avec les éléments suivants Services AWS pour le chiffrement, les notifications, la surveillance et l'audit :

  - AWS Key Management Service (AWS KMS)
  - Amazon Simple Notification Service (Amazon SNS)
  - Amazon CloudWatch : pour plus d'informations, consultezConfiguration des EventBridge règles pour les paramètres et des politiques de paramètres.
  - Amazon EventBridge : pour plus d'informations, consultez Surveillance des changements d'état du Systems Manager à l'aide des notifications Amazon SNS etRéférence : modèles et types d'événements Amazon EventBridge pour Systems Manager.

- AWS CloudTrail: Pour plus d'informations, consultez Journalisation des appels d' AWS Systems Manager API avec AWS CloudTrail.

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 d'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 Store prend en charge trois types de paramètres. `String`, `StringList` et `SecureString`.

À une exception près, lorsque vous créez ou mettez à jour un paramètre, vous saisissez sa valeur sous forme de texte brut et Parameter Store n'effectue aucune validation sur le texte en question. Pour les paramètres `String`, vous pouvez néanmoins spécifier le type de données comme `aws:ec2:image`. Alors, Parameter Store valide la valeur que vous saisissez comme étant au format approprié pour une AMI Amazon EC2 ; par exemple : `ami-12345abcdeEXAMPLE`.

Type de paramètre : String

Par défaut, les paramètres `String` sont constitués d'un bloc de texte que vous saisissez. Par exemple :

- `abc123`
- `Example Corp`
- `<img src="images/bannerImage1.png"/>`

Type de paramètre : 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
```

Type de paramètre : 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 Créer un paramètre SecureString (AWS CLI).

Nous recommandons l'utilisation des paramètres `SecureString` pour les scénarios suivants :

- Vous souhaitez utiliser les données/paramètres Services AWS sans exposer les valeurs sous forme de texte brut dans les commandes, les fonctions, les journaux des agents ou les journaux. CloudTrail
- Vous voulez contrôler les personnes ayant accès aux données sensibles.
- Vous souhaitez être en mesure d'auditer les accès à des données sensibles (CloudTrail).
- Vous voulez chiffrer vos données sensibles et vous voulez utiliser vos propres clés de chiffrement pour la gestion des accès.

> **Important**
>
> Seule la valeur d'un paramètre `SecureString` est chiffrée. Les noms de paramètres, les descriptions et d'autres propriétés ne sont pas chiffrés.

Vous pouvez utiliser le type de `SecureString` paramètre 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 que vous souhaitez protéger. `SecureString`les données sont cryptées et déchiffrées à l'aide d'une clé. AWS KMS Vous pouvez utiliser une clé KMS par défaut fournie par AWS ou créer et utiliser la vôtre AWS KMS key. (Utilisez votre propre AWS KMS key pour restreindre l'accès des utilisateurs aux paramètres `SecureString`. Pour de plus amples informations, veuillez consulter Autorisations IAM pour l'utilisation des clés AWS par défaut et des clés gérées par le client.)

Vous pouvez également utiliser des `SecureString` paramètres avec d'autres Services AWS. Dans l'exemple suivant, la fonction Lambda récupère un `SecureString` paramètre à l'aide de l'API. GetParameters

```
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
```

AWS KMS chiffrement et tarification

Si vous choisissez le type de `SecureString` paramètre lorsque vous créez votre paramètre, Systems Manager l'utilise AWS KMS pour chiffrer la valeur du paramètre.

> **Important**
>
> Parameter Store prend uniquement en charge des clés KMS à chiffrement symétrique. Vous ne pouvez pas utiliser une clé KMS à chiffrement asymétrique pour chiffrer vos paramètres.

Pour savoir si une clés KMS est symétrique ou asymétrique, consultez Identification de clés symétriques et asymétriques dans le Guide du développeur AWS Key Management Service .

La création d'Parameter Storeun `SecureString` paramètre est gratuite, mais des frais d'utilisation du AWS KMS chiffrement s'appliquent. Pour obtenir des informations, veuillez consulter Tarification AWS Key Management Service.

Pour plus d'informations sur les clés gérées par le client Clés gérées par AWS et les clés gérées par le client, consultez la section AWS Key Management Service Concepts du guide du AWS Key Management Service développeur. Pour plus d'informations sur le AWS KMS chiffrement Parameter Store et le chiffrement, consultez la section Comment AWS Systems ManagerParameter Store les utiliser AWS KMS.

Note

Pour afficher un Clé gérée par AWS, utilisez l' AWS KMS `DescribeKey`opération. Cet exemple AWS Command Line Interface (AWS CLI) est utilisé `DescribeKey` pour afficher un Clé gérée par AWS.

```
aws kms describe-key --key-id alias/aws/ssm
```

Plus d'informations

- Création d'un paramètre SecureString et association d'un nœud à un domaine (PowerShell)
- Parameter StoreÀ utiliser pour accéder en toute sécurité aux secrets et aux données de configuration dans CodeDeploy
- Articles intéressants sur Amazon EC2 Systems Manager Parameter Store

Configuration de Parameter Store

Avant de configurer les paramètres dans Parameter Store, une des fonctionnalités de AWS Systems Manager, vous devez configurer les politiques AWS Identity and Access Management (IAM) qui accordent aux utilisateurs de votre compte les autorisations nécessaires pour effectuer les actions que vous spécifiez. Cette section inclut des informations sur la façon de configurer manuellement ces politiques 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 politiques pour contrôler les actions de paramètres qui peuvent être exécutées sur un nœud géré. 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 invoquer 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 politiques IAM

Vous limitez l'accès aux AWS Systems Manager paramètres en utilisant AWS Identity and Access Management (IAM). Plus précisément, vous créez des politiques IAM qui restreignent l'accès aux opérations API suivantes :

- DeleteParameter
- DeleteParameters
- DescribeParameters
- GetParameter
- GetParameters
- GetParameterHistory
- GetParametersByPath
- PutParameter

Lorsque vous utilisez des politiques IAM pour restreindre l'accès aux paramètres Systems Manager, nous vous recommandons de créer et d'utiliser des politiques IAM restrictives. Par exemple, la politique 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-*"
        }
    ]
}
```

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 à un chemin `/a`, il peut également accéder à `/a/b`. Même si un utilisateur s'est vu refuser explicitement l'accès au paramètre `/a/b` dans IAM, il peut toujours appeler l'opération d'API `GetParametersByPath` de manière récursive pour `/a` et afficher `/a/b`.

Pour les administrateurs de confiance, vous pouvez fournir un accès complet à toutes les opérations d'API des paramètres Systems Manager en utilisant une politique similaire à l'exemple suivant. Cette politique accorde à l'utilisateur un accès complet à tous les paramètres de production qui commencent par `dbserver-prod-*`.

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

```
                "ssm:DeleteParameter",
                "ssm:GetParameterHistory",
                "ssm:GetParametersByPath",
                "ssm:GetParameters",
                "ssm:GetParameter",
                "ssm:DeleteParameters"
            ],
            "Resource": "arn:aws:ssm:us-east-2:123456789012:parameter/dbserver-prod-*"
        },
        {
            "Effect": "Allow",
            "Action": "ssm:DescribeParameters",
            "Resource": "*"
        }
    ]
}
```

Refuser des autorisations

Chaque API est unique et dispose d'opérations et d'autorisations distinctes que vous pouvez autoriser ou refuser individuellement. Un refus explicite dans n'importe quelle politique remplace l'autorisation.

> **Note**
>
> La clé par défaut AWS Key Management Service (AWS KMS) `Decrypt` autorise tous les principaux IAM au sein de. Compte AWS Si vous voulez disposer de différents niveaux d'accès aux paramètres `SecureString` dans votre compte, n'utilisez pas la clé par défaut.

Si vous voulez que toutes les opérations d'API qui récupèrent des valeurs de paramètres aient un comportement identique, vous pouvez utiliser un modèle tel que `GetParameter*` dans une politique. L'exemple suivant montre comment refuser `GetParameter`, `GetParameters`, `GetParameterHistory` et `GetParametersByPath` pour tous les paramètres commençant par `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 permettant à l'utilisateur d'en exécuter d'autres sur tous les paramètres commençant par `prod-*`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "ssm:PutParameter",
                "ssm:DeleteParameter",
                "ssm:DeleteParameters",
                "ssm:DescribeParameters"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetParametersByPath",
                "ssm:GetParameters",
                "ssm:GetParameter",
                "ssm:GetParameterHistory"
            ],
            "Resource": "arn:aws:ssm:us-east-2:123456789012:parameter/prod-*"
        }
    ]
}
```

> **Note**
>
> L'historique des paramètres inclut toutes les versions de paramètres, y compris la version actuelle. Par conséquent, si un utilisateur se voit refuser l'autorisation pour `GetParameter`, `GetParameters` et `GetParameterByPath`, mais qu'il obtient l'autorisation pour

`GetParameterHistory`, il peut voir le paramètre actuel, y compris les paramètres `SecureString`, en utilisant `GetParameterHistory`.

Autoriser uniquement l'exécution de paramètres spécifiques sur des nœuds

Vous pouvez contrôler l'accès afin que les nœuds gérés puissent uniquement exécuter les paramètres que vous spécifiez.

Si vous choisissez le type de `SecureString` paramètre lorsque vous créez votre paramètre, Systems Manager l'utilise AWS KMS pour chiffrer la valeur du paramètre. AWS KMS chiffre la valeur à l'aide d'une clé gérée par le client Clé gérée par AWS ou d'une clé gérée par le client. Pour plus d'informations sur AWS KMS et AWS KMS key, consultez le Guide du AWS Key Management Service développeur.

Vous pouvez les consulter Clé gérée par AWS en exécutant la commande suivante à partir du AWS CLI.

```
aws kms describe-key --key-id alias/aws/ssm
```

L'exemple suivant permet aux nœuds d'obtenir une valeur de paramètre seulement pour les paramètres commençant par `prod-`. Si le paramètre est un paramètre `SecureString`, le nœud déchiffre alors la chaîne en utilisant la AWS KMS.

> **Note**
>
> Les politiques 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 politiques aux utilisateurs et aux instances, consultez Utilisation de Systems Manager avec des instances EC2.

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

```
            ],
            "Resource": [
                "arn:aws:ssm:us-east-2:123456789012:parameter/prod-*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": [
                "arn:aws:kms:us-east-2:123456789012:key/4914ec06-e888-4ea5-
a371-5b88eEXAMPLE"
            ]
        }
    ]
}
```

Autorisations IAM pour l'utilisation des clés AWS par défaut et des clés gérées par le client

Parameter Store`SecureString`les paramètres sont chiffrés et déchiffrés à l'aide de clés. AWS KMS Vous pouvez choisir de chiffrer vos `SecureString` paramètres à l'aide d'une clé KMS AWS KMS key ou de la clé KMS par défaut fournie par AWS.

Lorsque vous utilisez une clé gérée par le client, la politique IAM qui accorde à un utilisateur l'accès à un paramètre ou à un chemin d'accès doit fournir des autorisations `kms:Encrypt` explicites pour la clé. Par exemple, la politique suivante permet à un utilisateur de créer, de mettre à jour et d'afficher des `SecureString` paramètres commençant `prod-` par le Région AWS et spécifié Compte AWS.

```
{
    "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 par défaut gérée par AWS . Si vous utilisez cette clé par défaut pour chiffrer des paramètres `SecureString` et que vous ne souhaitez pas que les utilisateurs utilisent des paramètres `SecureString`, leurs politiques IAM doivent explicitement refuser l'accès à la clé par défaut, comme illustré dans l'exemple de politique suivant.

> **Note**
>
> Vous pouvez localiser l'Amazon Resource Name (ARN) de la clé par défaut dans la console AWS KMS sur la page 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": [
                "arn:aws:kms:us-east-2:111122223333:key/abcd1234-ab12-cd34-ef56-
abcdeEXAMPLE"
            ]
        }
    ]
}
```

Si vous avez besoin d'un contrôle d'accès précis sur les paramètres `SecureString` de votre compte, vous devez utiliser une clé gérée par le client pour protéger et restreindre l'accès à ces paramètres. Nous vous recommandons également de l'utiliser AWS CloudTrail pour surveiller l'activité des `SecureString` paramètres.

Pour plus d'informations, consultez les rubriques suivantes :

- Logique d'évaluation des politiques dans le Guide de l'utilisateur IAM
- Utilisation de politiques de clé dans AWS KMS dans le Guide du développeur AWS Key Management Service
- Afficher les événements avec l'historique des CloudTrail événements dans le guide de AWS CloudTrail l'utilisateur

Gestion des niveaux de paramètres

Parameter Store, une fonctionnalité de AWS Systems Manager, 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 retour d'un paramètre avancé à un paramètre standard entraînerait une réduction de la taille du paramètre de 8 Ko à 4 Ko, ce qui provoquerait une perte de données. Cela entraînerait également la suppression de toutes les politiques 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 AWS KMS dans le Guide du développeur AWS Key Management Service .

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	Avancé
Nombre total de paramètres autorisés (par Compte AWS et Région AWS)	10 000	100 000
Taille maximale d'une valeur de paramètre.	4 Ko	8 Ko
Politiques de paramètre disponibles	Non	Oui Pour plus d'informations, consultez Affectation de politiques de paramètres.
Coût	Pas de frais supplémentaires	Des frais supplémentaires seront facturés Pour plus d'informations, consultez la section AWS Systems Manager Tarification pour Parameter Store.

Rubriques

- Spécification d'un niveau de paramètre par défaut
- Remplacement d'un paramètre standard par un paramètre avancé

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'opération `PutParameter`), vous pouvez spécifier le niveau de paramètre à utiliser dans la demande. Voici un exemple avec utilisation de l' AWS Command Line Interface (AWS CLI).

Linux & macOS

```
aws ssm put-parameter \
    --name "default-ami" \
    --type "String" \
    --value "t2.micro" \
    --tier "Standard"
```

Windows

```
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 :

- Advanced (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 AWS CloudFormation modèles fournissent de nombreux paramètres lorsqu'ils sont exécutés. Lorsque ce processus vous amène à atteindre le quota de 10 000 paramètres dans le niveau de paramètres standard, Intelligent-Tiering vous fait automatiquement passer au niveau de paramètres avancés et vos processus ne sont pas interrompus. AWS CloudFormation
- Vous stockez une valeur de certificat dans un paramètre, effectuez une rotation régulière de la valeur de certificat et le contenu est inférieur au quota 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 politique de paramètre, ce qui nécessite le niveau paramètre avancé. Au lieu d'inclure dans tous les appels l'option `--tier Advanced` de mise 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 politique de paramètre.
- Plus de 10 000 paramètres existent déjà Compte AWS dans votre compte actuel Région AWS.

Options de niveau par défaut

Les options de niveau que vous pouvez spécifier comme options par défaut sont les suivantes.

- Standard – le niveau de paramètre standard est le niveau par défaut lorsque vous commencez à utiliser Parameter Store. À l'aide du niveau de paramètres standard, vous pouvez créer 10 000

paramètres pour chacun d'entre eux Région AWS 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 politiques de paramètre. L'utilisation du niveau paramètre standard n'entraîne pas de frais supplémentaires. Si vous sélectionnez 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 : utilisez le niveau de paramètres avancés pour créer un maximum de 100 000 paramètres pour chacun Région AWS d'entre eux 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 politiques de paramètre. L'utilisation du niveau paramètre avancé est facturée. Pour plus d'informations, consultez la section AWS Systems Manager Tarification pour Parameter Store. Si vous sélectionnez 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 sélectionnez 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 dont le contenu est inférieur à 4 Ko, que le contenu actuel Région AWS de votre Compte AWS fichier contient moins de 10 000 paramètres et que vous ne spécifiez pas de politique de paramètres, un paramètre standard est créé. Si vous exécutez une commande pour créer un paramètre contenant plus de 4 Ko de contenu, si vous avez déjà plus de 10 000 paramètres en cours Région AWS dans votre fichier Compte AWS, ou si vous spécifiez une politique de paramètres, un paramètre avancé est créé.

  > **Note**
  >
  > Lorsque vous choisissez Intelligent-Tiering, autorisez explicitement AWS à débiter votre compte pour tous les paramètres avancés que vous avez 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 êtes autorisé dans AWS Identity and Access Management (IAM) à modifier le niveau des paramètres par défaut en Parameter Store effectuant l'une des opérations suivantes :

- Veillez à bien attacher la politique `AdministratorAccess` à votre entité IAM (qui peut être un utilisateur, un groupe ou un rôle).
- Assurez-vous que vous avez l'autorisation de modifier le paramètre de niveau par défaut à l'aide des opérations d'API suivantes :
  - GetServiceSetting
  - UpdateServiceSetting
  - ResetServiceSetting

Accordez les autorisations suivantes à l'entité IAM pour permettre aux utilisateurs d'afficher et de modifier le paramètre de niveau par défaut pour la configuration d'une Région AWS spécifique dans un Compte AWS.

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

Les administrateurs peuvent spécifier une autorisation en lecture seule en affectant les autorisations suivantes.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetServiceSetting"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Deny",
            "Action": [
                "ssm:ResetServiceSetting",
                "ssm:UpdateServiceSetting"
            ],
            "Resource": "*"
        }
    ]
}
```

Pour activer l'accès, ajoutez des autorisations à vos utilisateurs, groupes ou rôles :

- Utilisateurs et groupes dans AWS IAM Identity Center :

  Créez un jeu d'autorisations. Suivez les instructions de la rubrique Création d'un jeu d'autorisations du Guide de l'utilisateur AWS IAM Identity Center .

- Utilisateurs gérés dans IAM par un fournisseur d'identité :

  Créez un rôle pour la fédération d'identité. Pour plus d'informations, voir la rubrique Création d'un rôle pour un fournisseur d'identité tiers (fédération) du Guide de l'utilisateur IAM.

- Utilisateurs IAM :
  - Créez un rôle que votre utilisateur peut assumer. Suivez les instructions de la rubrique Création d'un rôle pour un utilisateur IAM du Guide de l'utilisateur IAM.
  - (Non recommandé) Attachez une politique directement à un utilisateur ou ajoutez un utilisateur à un groupe d'utilisateurs. Suivez les instructions de la rubrique Ajout d'autorisations à un utilisateur (console) du Guide de l'utilisateur IAM.

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 paramètres par défaut pour le Compte AWS et actuel Région AWS.

> **Tip**
>
> Si vous n'avez pas encore créé de paramètre, vous pouvez utiliser le AWS Command Line Interface (AWS CLI) ou AWS Tools for 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) et Spécifier ou modifier le niveau Parameter Store par défaut (PowerShell).

Pour spécifier ou modifier le niveau Parameter Store par défaut

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Parameter Store.
3. Sélectionnez l'onglet Settings.
4. Sélectionnez Change default tier (Modifier le niveau par défaut).
5. Sélectionnez l'une des options suivantes :

   - Standard
   - Advanced (Avancé)
   - Intelligent-Tiering (Hiérarchisation intelligente)

   Pour de plus amples informations sur ces options, consultez Spécification d'un niveau de paramètre par défaut.
6. Examinez le message, puis sélectionnez 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 le AWS CLI pour modifier le paramètre de niveau par défaut pour le paramètre actuel Compte AWS et Région AWS.

Pour spécifier ou modifier le niveau Parameter Store par défaut à l'aide de l' AWS CLI

1. Ouvrez le AWS CLI et exécutez la commande suivante pour modifier le paramètre de niveau par défaut pour un paramètre spécifique Région AWS 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 Région AWS prise en charge par AWS Systems Manager, par exemple `us-east-2` pour la région USA Est (Ohio). Pour obtenir une liste des valeurs *region* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la Référence générale d'Amazon Web Services.

   Les valeurs *tier-option* incluent `Standard`, `Advanced` et `Intelligent-Tiering`. Pour de plus amples informations sur ces options, consultez Spécification d'un niveau de paramètre par défaut.

   Il n'y a pas de sortie si la commande réussit.
2. Exécutez la commande suivante pour afficher les paramètres de service actuels du niveau des paramètres par défaut pour Parameter Store les versions actuelles Compte AWS et Région AWS.

   ```
   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`.

Spécifier ou modifier le niveau Parameter Store par défaut (PowerShell)

La procédure suivante explique comment utiliser les Outils pour Windows PowerShell afin de modifier le paramètre par défaut défini pour un élément spécifique Région AWS d'un compte Amazon Web Services.

Pour spécifier ou modifier le niveau Parameter Store par défaut à l'aide de PowerShell

1. Modifiez le niveau Parameter Store par défaut dans le niveau actuel Compte AWS et Région AWS en utilisant le AWS Tools for PowerShell (Outils 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 Région AWS prise en charge par AWS Systems Manager, par exemple `us-east-2` pour la région USA Est (Ohio). Pour obtenir une liste des valeurs *region* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la Référence générale d'Amazon Web Services.

   Les valeurs *tier-option* incluent `Standard`, `Advanced` et `Intelligent-Tiering`. Pour de plus amples informations sur ces options, consultez Spécification d'un niveau de paramètre par défaut.

   Il n'y a pas de sortie si la commande réussit.

2. Exécutez la commande suivante pour afficher les paramètres de service actuels du niveau des paramètres par défaut pour Parameter Store les versions actuelles Compte AWS et Région AWS.

```
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 Région AWS prise en charge par AWS Systems Manager, par exemple `us-east-2` pour la région USA Est (Ohio). Pour obtenir une liste des valeurs *region* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la 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, consultez Création de paramètres Systems Manager.

Pour remplacer un paramètre standard par un paramètre avancé

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Parameter Store.
3. Sélectionnez un paramètre, puis sélectionnez Edit (Modifier).
4. Pour Description, entrez les informations concernant ce paramètre.
5. Choisir Advanced (Avancé).
6. Pour Value (Valeur), saisissez la valeur de ce paramètre. La valeur maximale des paramètres avancés est de 8 Ko.

7. Sélectionnez Enregistrer les modifications.

Augmenter ou réinitialiser le débit Parameter Store

L'augmentation du Parameter Store débit augmente le nombre maximal de transactions par seconde (TPS) quiParameter Store, grâce à la capacité de AWS Systems Manager, peuvent être traitées. 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 plusieurs paramètres. Vous pouvez augmenter le quota jusqu'au débit maximal sous l'onglet Settings (Paramètres).

Pour plus d'informations sur le débit maximal par défaut et les limites maximales, consultez la section AWS Systems Manager Points de terminaison et quotas.

L'augmentation du quota de débit entraîne des frais pour votre. Compte AWS Pour plus d'informations, consultez Tarification d'AWS Systems Manager.

> **Note**
>
> Le paramètre de Parameter Store débit s'applique à toutes les transactions créées par tous les utilisateurs IAM dans les versions actuelles Compte AWS et. Région AWS Le paramètre de débit s'applique aux paramètres standard et avancés.

Rubriques

- Configuration des autorisations pour modifier le Parameter Store débit
- Augmenter ou réinitialiser le débit (console)
- Augmenter ou réinitialiser le débit ()AWS CLI
- Augmenter ou réinitialiser le débit () PowerShell

Configuration des autorisations pour modifier le Parameter Store débit

Vérifiez que vous êtes autorisé dans IAM à modifier le Parameter Store débit en effectuant l'une des opérations suivantes :

- Veillez à bien attacher la politique `AdministratorAccess` à votre entité IAM (qui peut être un utilisateur, un groupe ou un rôle).

- Assurez-vous que vous avez l'autorisation de modifier le paramètre de service de débit à l'aide des opérations d'API suivantes :
  - GetServiceSetting
  - UpdateServiceSetting
  - ResetServiceSetting

Accordez les autorisations suivantes à l'entité IAM pour permettre à un utilisateur d'afficher et de modifier le paramètre parameter-throughput pour les paramètres d'une Région AWS spécifique dans un Compte AWS.

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

Les administrateurs peuvent spécifier une autorisation en lecture seule en affectant les autorisations suivantes.

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

```
                "ssm:GetServiceSetting"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Deny",
            "Action": [
                "ssm:ResetServiceSetting",
                "ssm:UpdateServiceSetting"
            ],
            "Resource": "*"
        }
    ]
}
```

Pour activer l'accès, ajoutez des autorisations à vos utilisateurs, groupes ou rôles :

- Utilisateurs et groupes dans AWS IAM Identity Center :

  Créez un jeu d'autorisations. Suivez les instructions de la rubrique Création d'un jeu d'autorisations du Guide de l'utilisateur AWS IAM Identity Center .

- Utilisateurs gérés dans IAM par un fournisseur d'identité :

  Créez un rôle pour la fédération d'identité. Pour plus d'informations, voir la rubrique Création d'un rôle pour un fournisseur d'identité tiers (fédération) du Guide de l'utilisateur IAM.

- Utilisateurs IAM :
  - Créez un rôle que votre utilisateur peut assumer. Suivez les instructions de la rubrique Création d'un rôle pour un utilisateur IAM du Guide de l'utilisateur IAM.
  - (Non recommandé) Attachez une politique directement à un utilisateur ou ajoutez un utilisateur à un groupe d'utilisateurs. Suivez les instructions de la rubrique Ajout d'autorisations à un utilisateur (console) du Guide de l'utilisateur IAM.

Augmenter ou réinitialiser le 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 les Compte AWS et Région AWS actuels. Il indique également comment revenir aux paramètres standard si vous n'avez plus besoin d'augmenter le débit ou si vous ne souhaitez plus encourir de frais.

> **Tip**
>
> Si vous n'avez pas encore créé de paramètre, vous pouvez utiliser le AWS Command Line Interface (AWS CLI) ou AWS Tools for Windows PowerShell pour augmenter le débit. Pour plus d'informations, consultez Augmenter ou réinitialiser le débit ()AWS CLI et Augmenter ou réinitialiser le débit () PowerShell.

Pour augmenter ou réinitialiser le Parameter Store débit

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Parameter Store.
3. Sélectionnez l'onglet Settings.
4. Pour augmenter le débit, choisissez Définir la limite.

   -ou-

   Pour revenir à la limite par défaut, choisissez Réinitialiser la limite.
5. Si vous augmentez la limite, procédez comme suit :

   - Cochez la case J'accepte que la modification de ce paramètre entraîne des frais sur mon Compte AWS compte.
   - Sélectionnez Set limit (Définir la limite).

   -ou-

   Si vous redéfinissez la limite par défaut, procédez comme suit :

   - Cochez la case J'accepte que la réinitialisation à la limite de débit par défaut entraîne le traitement Parameter Store de moins de transactions par seconde.
   - Choisissez Réinitialiser la limite.

Augmenter ou réinitialiser le débit ()AWS CLI

La procédure suivante montre comment utiliser le AWS CLI pour augmenter le nombre de transactions par seconde Parameter Store pouvant être traitées pour le Compte AWS et actuel Région AWS. Vous pouvez également revenir à la limite par défaut.

Pour augmenter le Parameter Store débit à l'aide du AWS CLI

1. Ouvrez le AWS CLI et exécutez la commande suivante pour augmenter le nombre de transactions par seconde Parameter Store pouvant être traitées dans le Compte AWS et actuel Région AWS.

   ```
   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 actuels du service de débit pour Parameter Store les versions actuelles Compte AWS et Région AWS.

   ```
   aws ssm get-service-setting --setting-id arn:aws:ssm:region:account-
   id:servicesetting/ssm/parameter-store/high-throughput-enabled
   ```

   Le système renvoie des informations similaires à ce qui suit :

   ```
   {
       "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"
    }
}
```

Augmenter ou réinitialiser le débit () PowerShell

La procédure suivante montre comment utiliser les Outils pour Windows PowerShell afin d'augmenter le nombre de transactions par seconde Parameter Store pouvant être traitées pour le Compte AWS et actuel Région AWS. Vous pouvez également revenir à la limite par défaut.

Pour augmenter le Parameter Store débit en utilisant PowerShell

1. Augmentez le Parameter Store débit dans le courant Compte AWS et Région AWS en utilisant les AWS Tools for PowerShell (Outils pour PowerShell).

   ```
   Update-SSMServiceSetting -SettingId "arn:aws:ssm:region:account-id:servicesetting/
   ssm/parameter-store/high-throughput-enabled" -SettingValue "true" -Region region
   ```

   Il n'y a pas de sortie si la commande réussit.

2. Exécutez la commande suivante pour afficher les paramètres actuels du service de débit pour Parameter Store les versions actuelles Compte AWS et Région AWS.

   ```
   Get-SSMServiceSetting -SettingId "arn:aws:ssm:region:account-id:servicesetting/ssm/
   parameter-store/high-throughput-enabled" -Region region
   ```

Le système renvoie des informations similaires à ce qui suit :

```
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
```

Le système renvoie des informations similaires à ce qui suit :

```
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 vous informer des modifications apportées aux paramètres. AWS Systems Manager Vous pouvez créer une EventBridge règle pour vous avertir lorsqu'un paramètre ou une version d'étiquette de paramètre est créé, mis à jour ou supprimé. Les événements sont générés dans la mesure du possible. Vous pouvez être informé des changements ou du statut lié aux politiques de paramètre, par exemple lorsqu'un paramètre expire, va expirer ou n'a pas changé pendant un laps de temps spécifié.

> **Note**
>
> Les politiques de paramètre sont disponibles pour les paramètres qui utilisent le niveau de paramètres avancés. Des frais supplémentaires seront facturés. Pour plus d'informations, consultez Affectation de politiques de paramètres et Gestion des niveaux de paramètres.

Les rubriques de cette section expliquent également comment lancer 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 invoquer une fonction Lambda lorsque le mot de passe de votre base de données est mis à jour. La fonction Lambda peut forcer les connexions de votre base de données à réinitialiser ou à vous reconnecter avec le nouveau mot de passe. EventBridge prend également en charge l'exécution de Run Command commandes et les exécutions automatisées, ainsi que les actions dans de nombreux autres domainesServices AWS. Run Commandet l'automatisation sont toutes deux des fonctionnalités deAWS Systems Manager. Pour plus d'informations, consultez le guide de EventBridge l'utilisateur Amazon.

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 plus d'informations, consultez Prise en main d'Amazon SNS dans le Guide du développeur Amazon Simple Notification Service.

Configuration des EventBridge règles pour les paramètres et des politiques de paramètres

Cette rubrique explique les sections suivantes :

- Comment créer une EventBridge règle qui invoque une cible en fonction d'événements survenus dans un ou plusieurs paramètres de votreCompte AWS.
- Comment créer des EventBridge règles qui invoquent des cibles en fonction d'événements liés à une ou plusieurs politiques de paramètres de votreCompte AWS. Lorsque vous créez un paramètre, vous spécifiez le moment où un paramètre expire, le moment où recevoir une notification avant qu'un paramètre n'expire, ainsi que le délai d'attente avant l'envoi d'une notification indiquant qu'un paramètre n'a pas changé. Procédez comme suit pour configurer la notification pour ces événements. Pour plus d'informations, consultez Affectation de politiques de paramètres et Gestion des niveaux de paramètres.

Pour configurer une EventBridge règle pour un paramètre ou une politique de paramètres de Systems Manager

1. Ouvrez la EventBridge console Amazon à l'adresse https://console.aws.amazon.com/events/.
2. Dans le volet de navigation, sélectionnez Rules (Règles), puis Create rule (Créer une règle).

   -ou-

   Si la page d' EventBridge 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 région et sur le même bus d'événement.
4. Pour Event bus (Bus d'événement), sélectionnez le bus d'événement que vous souhaitez associer à cette règle. Si vous souhaitez que cette règle s'applique aux événements correspondants qui proviennent de votre propre Compte AWS, sélectionnez default, (par défaut). Lorsqu'un Service AWS de votre compte émet un événement, il accède toujours au bus d'événement par défaut de votre compte.
5. Pour Rule type (Type de règle), ne désactivez pas la valeur par défaut Rule with an event pattern (Règle avec un modèle d'événement).
6. Choisissez Suivant.
7. Pour Source d'événement, laissez les AWSévénements par défaut ou les événements EventBridge partenaires sélectionnés. Vous pouvez ignorer la section Sample event (Exemple d'événement).
8. Pour Event pattern (Modèle d'événement), procédez comme suit :

   - Choisissez Custom pattern (JSON editor) (Modèle personnalisé [éditeur JSON]).
   - Pour Event pattern (Modèle d'événement), collez l'un des contenus suivants dans la zone, selon que vous créez une règle pour un paramètre ou une politique de paramètres :

     Parameter

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

Parameter policy

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

- Modifiez le contenu pour les paramètres et les opérations sur lesquels vous voulez agir, comme le montrent les exemples suivants.

Parameter

Avec cet exemple, une action est entreprise lorsque l'un des paramètres nommés /Oncall ou /Project/Teamlead est mis à jour :

```
{
    "source": [
        "aws.ssm"
    ],
    "detail-type": [
        "Parameter Store Change"
    ],
    "detail": {
        "name": [
            "/Oncall",
            "/Project/Teamlead"
        ],
        "operation": [
            "Update"
        ]
    }
}
```

Parameter policy

Avec cet exemple, une action est entreprise 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"
        ]
```

```
    }
}
```

9. Choisissez Suivant.
10. Pour Target 1 (Cible 1), sélectionnez un type de cible et une ressource prise en charge. Par exemple, si vous sélectionnez SNS topic (Rubrique SNS), sélectionnez une valeur pour Topic (Rubrique). Si vous le souhaitez CodePipeline, entrez un ARN de pipeline pour l'ARN de pipeline. Fournissez des valeurs de configuration supplémentaires au besoin.

> **Tip**
>
> Choisissez Add another target (Ajouter une autre cible) si vous avez besoin de cibles supplémentaires pour la règle.

11. Choisissez Suivant.
12. (Facultatif) Saisissez une ou plusieurs balises pour la règle. Pour plus d'informations, consultez les EventBridgebalises Amazon dans le guide de EventBridge l'utilisateur Amazon.
13. Choisissez Suivant.
14. Choisissez Créer une règle.

Plus d'informations

- Utiliser les étiquettes de paramètre pour faciliter la mise à jour de la configuration entre les environnements
- Tutoriel : EventBridge à utiliser pour relayer des événements AWS Systems ManagerRun Command dans le guide de EventBridge l'utilisateur Amazon
- Tutoriel : définissez AWS Systems Manager l'automatisation comme EventBridge objectif dans le guide de EventBridge l'utilisateur Amazon

Utilisation de l'option Parameter Store

Cette section décrit comment organiser et créer des paramètres de balises, et créer différentes versions de paramètres. Vous pouvez utiliser la AWS Systems Manager console, la console Amazon Elastic Compute Cloud (Amazon EC2) ou AWS Command Line Interface le AWS CLI () pour créer et utiliser des paramètres. Pour de plus amples informations sur les paramètres , veuillez consulter Qu'est-ce qu'un paramètre ?

Rubriques

Création de paramètres Systems Manager

Utilisez les informations des rubriques suivantes pour créer des paramètres Systems Manager via la console AWS Systems Manager, la AWS Command Line Interface (AWS CLI) ou AWS Tools for Windows PowerShell (Tools for Windows PowerShell).

Cette section explique comment créer, stocker et exécuter des paramètres avec Parameter Store dans un environnement de test. Il explique également comment utiliser Parameter Store avec d'autres fonctionnalités Systems Manager et Services AWS. Pour plus d'informations, reportez-vous à Qu'est-ce qu'un paramètre ?

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 indiquées dans la rubrique PutParameter de la 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 de paramètre sont sensibles à la casse.
- Espaces : les noms de paramètre ne peuvent pas comporter d'espaces.

- Caractères valides : les noms de paramètre ne peuvent inclure que les lettres et symboles suivants : `a-zA-Z0-9_.-`

  De plus, le caractère oblique (/) est utilisé pour délimiter les hiérarchies de noms de paramètres. Par exemple : `/Dev/Production/East/Project-ABC/MyParameter`

- Valid AMI format (Format d'AMI valide) : lorsque vous sélectionnez `aws:ec2:image` comme type de données pour un paramètre `String`, l'ID que vous saisissez doit être valide pour le format d'ID d'AMI `ami-12345abcdeEXAMPLE`.

- Fully qualified (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 paramètres complets : `MyParameter1`, `/MyParameter2`, `/Dev/Production/East/Project-ABC/MyParameter`
  - Nom du paramètre non complet : `MyParameter3/L1`

- Longueur : la longueur maximale pour le nom du paramètre que vous créez est de 1011 caractères. Cela inclut les caractères de l'ARN qui précèdent le nom que vous spécifiez, tels que `arn:aws:ssm:us-east-2:111122223333:parameter/`.

- Préfixes : le préfixe « `aws` » ou « `ssm` » (non sensible à la casse) n'est pas autorisé pour un nom de paramètre. Par exemple, les tentatives de création de paramètres portant 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. Par exemple, {{ssm:*parameter-name*}} et {{ ssm:*parameter-name* }}, comme `{{ssm:MyParameter}}` et `{{ ssm:MyParameter }}`.

- Uniqueness (Unicité) : un nom de paramètre doit être unique dans une Région AWS. 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 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 à un chemin `/a`, il peut également accéder à `/a/b`. Même si un utilisateur s'est vu refuser explicitement l'accès au paramètre `/a/b` dans AWS Identity and Access Management (IAM), il peut toujours appeler l'opération d'API GetParametersByPath de manière récursive pour `/a` et afficher `/a/b`.

Rubriques

- Créer un paramètre Systems Manager (console)
- Créer un paramètre Systems Manager (AWS CLI)
- Créer un paramètre Systems Manager (Tools for Windows PowerShell)

Créer un paramètre Systems Manager (console)

Vous pouvez utiliser la AWS Systems Manager console pour créer et exécuter `String` `StringList` des types de `SecureString` paramètres. Après avoir supprimé un paramètre, attendez au moins 30 secondes avant de créer un paramètre avec le même nom.

> **Note**
>
> Les paramètres ne sont disponibles que Région AWS là où ils ont été créés.

La procédure suivante vous guide à travers le processus de création d'un paramètre à l'aide de la console Parameter Store. Vous pouvez créer des types de paramètres `String`, `StringList` et `SecureString` à partir de la console.

Pour créer un paramètre

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Parameter Store.
3. Sélectionnez Create parameter.
4. Dans la zone Name (Nom), entrez une hiérarchie et un nom. Par exemple, saisissez **`/Test/helloWorld`**.

   Pour plus d'informations sur les hiérarchies de paramètres, consultez Utiliser des hiérarchies de paramètres.
5. Dans la zone Description, entrez une description qui identifie ce paramètre comme paramètre de test.
6. Pour Parameter tier (Niveau du paramètre), sélectionnez Standard ou Advanced (Avancé). Pour de plus amples informations sur les paramètres avancés, veuillez consulter Gestion des niveaux de paramètres.
7. Pour Type, choisissez String StringList, ou SecureString.

   - Si vous sélectionnez 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 le souhaitez SecureString, le champ KMS Key ID s'affiche. Si vous ne fournissez pas d' AWS Key Management Service AWS KMS key ID, d' AWS KMS key Amazon Resource Name (ARN), de nom d'alias ou d'alias ARN, le système utilise`alias/aws/ssm`, à savoir le nom Clé gérée par AWS de Systems Manager. Si vous ne souhaitez pas utiliser cette clé, vous pouvez utiliser une clé gérée par le client. Pour de plus amples informations sur Clés gérées par AWS et les clés gérées par le client, reportez-vous à la section Concepts AWS Key Management Service dans le Guide du développeur AWS Key Management Service . Pour plus d'informations sur le AWS KMS chiffrement Parameter Store et le chiffrement, consultez la section Comment AWS Systems ManagerParameter Store les utiliser AWS KMS.

  > **Important**
  >
  > Parameter Store prend uniquement en charge des clés KMS à chiffrement symétrique. Vous ne pouvez pas utiliser une clé KMS à chiffrement asymétrique pour chiffrer vos paramètres. Pour savoir si une clés KMS est symétrique ou asymétrique, consultez Identification de clés symétriques et asymétriques dans le Guide du développeur AWS Key Management Service .

- Lorsque vous créez un paramètre `SecureString` dans la console à l'aide du paramètre `key-id` avec un nom d'alias de clé 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/abcd1234-ab12-cd34-ef56-abcdeEXAMPLE
  ```

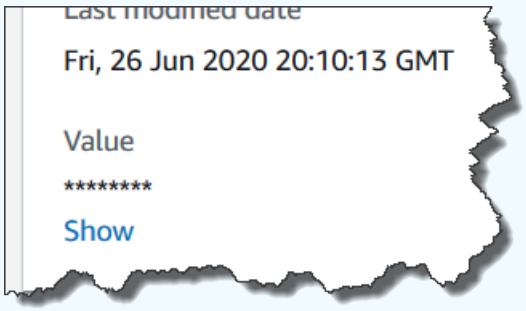
  Voici un exemple de nom d'alias.

  ```
  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 le souhaitez SecureString, la valeur du paramètre est masquée par défaut (« ****** ») lorsque vous la visualiserez ultérieurement dans l'onglet Aperçu des paramètres. Sélectionnez Show (Montrer) 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. Sélectionnez Create parameter.
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`, sélectionnez 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 de paramètres `String`, `StringList` et `SecureString`. Après avoir supprimé un paramètre, attendez au moins 30 secondes avant de créer un paramètre avec le même nom.

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.

> **Note**
>
> Les paramètres sont uniquement disponibles dans la Région AWS où ils ont été créés.

Rubriques

- Créer un paramètre String (AWS CLI)
- Créer un paramètre StringList (AWS CLI)
- Créer un paramètre SecureString (AWS CLI)
- Créer un paramètre multi-ligne (AWS CLI)

Créer un paramètre **`String`** (AWS CLI)

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS Command Line Interface (AWS CLI).

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l'AWS CLI.

2. Exécutez la commande suivante pour créer un paramètre de type `String`. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations.

   Linux & macOS

   ```
   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'Amazon Machine Image (AMI) comme valeur de paramètre.

Linux & macOS

```
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 Utiliser des hiérarchies de paramètres.

L'option `--data-type` doit être spécifiée uniquement si vous créez un paramètre contenant un ID d'AMI. Cela valide le fait que la valeur de paramètre saisie 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 la spécification d'une valeur n'est pas obligatoire. Pour de

plus amples informations, veuillez consulter Prise en charge de paramètres natifs pour les ID d'Amazon Machine Image.

> **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 de plus amples informations, veuillez consulter Prise en charge de paramètres natifs pour les ID d'Amazon Machine Image.

Dans l'exemple suivant, on ajoute deux balises de paires clé-valeur à un paramètre.

Linux & macOS

```
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 noms de paramètres pour créer un paramètre `String` en texte clair. Cela renvoie le numéro de version du paramètre. Pour plus d'informations sur les hiérarchies de paramètres, consultez Utiliser des hiérarchies de paramètres.

Linux & macOS

Paramètre non présent dans une hiérarchie

```
aws ssm put-parameter \
    --name "golden-ami" \
    --type "String" \
    --value "ami-12345abcdeEXAMPLE"
```

Paramètre présent dans une hiérarchie

```
aws ssm put-parameter \
    --name "/amis/linux/golden-ami" \
    --type "String" \
    --value "ami-12345abcdeEXAMPLE"
```

Windows

Paramètre non présent dans une hiérarchie

```
aws ssm put-parameter ^
    --name "golden-ami" ^
    --type "String" ^
    --value "ami-12345abcdeEXAMPLE"
```

Paramètre présent 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 de paramètre et vérifier les détails de votre nouveau paramètre.

```
aws ssm get-parameters --names "/Test/IAD/helloWorld"
```

Le système retourne des informations telles que les suivantes.

```
{
    "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. Cela 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 uniquement pour récupérer la valeur de paramètre.

```
aws ssm get-parameter --name testDataTypeParameter --query "Parameter.Value"
```

Exécutez la commande suivante uniquement pour récupérer la valeur de paramètre avec `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 retourne des informations telles que les suivantes.

```
{
    "Parameters": [
        {
            "Name": "helloworld",
            "Type": "String",
            "LastModifiedUser": "arn:aws:iam::123456789012:user/JohnDoe",
            "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 Command Line Interface (AWS CLI).

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l'AWS CLI.
2. Exécutez la commande suivante pour créer un paramètre. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations.

   Linux & macOS

   ```
   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 de votre ordinateur local, exécutez l'une des commandes suivantes pour télécharger et exécuter un script à partir de la version Windows 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 & macOS

```
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. Par 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`. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations.

> **Important**
>
> Seule la valeur d'un paramètre `SecureString` est chiffrée. Les noms de paramètres, les descriptions et d'autres propriétés ne sont pas chiffrés.

> **Important**
>
> Parameter Store prend uniquement en charge des clés KMS à chiffrement symétrique. Vous ne pouvez pas utiliser une clé KMS à chiffrement asymétrique pour chiffrer vos paramètres. Pour savoir si une clés KMS est symétrique ou asymétrique, consultez Identification de clés symétriques et asymétriques dans le Guide du développeur AWS Key Management Service.

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS Command Line Interface (AWS CLI).

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l'AWS CLI.

2. Exécutez l'une des commandes suivantes pour créer un paramètre utilisant le type de données `SecureString`.

   Linux & macOS

   Créer un paramètre **`SecureString`** à l'aide de la Clé gérée par AWS par défaut

```
aws ssm put-parameter \
    --name "parameter-name" \
    --value "parameter-value" \
    --type "SecureString"
```

Créer un paramètre **`SecureString`** utilisant une clé 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 paramètre **`SecureString`** utilisant une clé AWS KMS personnalisée

```
aws ssm put-parameter \
    --name "parameter-name" \
    --value "a-parameter-value, for example P@ssW%rd#1" \
    --type "SecureString" \
    --key-id "your-account-ID/the-custom-AWS KMS-key" \
    --tags "Key=tag-key,Value=tag-value"
```

Windows

Créer un paramètre **`SecureString`** à l'aide de la Clé gérée par AWS par défaut

```
aws ssm put-parameter ^
    --name "parameter-name" ^
    --value "parameter-value" ^
    --type "SecureString"
```

Créer un paramètre **`SecureString`** utilisant une clé 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 paramètre **`SecureString`** utilisant une clé AWS KMS 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"account-ID/the-custom-AWS KMS-key"
```

Si vous créez un paramètre `SecureString` à l'aide de la clé Clé 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 AWS KMS key par défaut affectée à votre Compte AWS et Région AWS, supprimez le paramètre `key-id` à partir de la commande. Pour plus d'informations sur la configuration d'une règle dans AWS KMS keys, consultez les Concepts AWS Key Management Service dans le Guide du développeur AWS Key Management Service.

Pour utiliser une clé gérée par le client au lieu de la Clé gérée par AWS affectée à votre compte, spécifiez la clé à l'aide du paramètre `--key-id`. Le paramètre prend en charge les formats de paramètre KMS suivants.

- Exemple d'Amazon Resource Name (ARN) de clé :

  `arn:aws:kms:us-east-2:123456789012:key/key-id`

- Exemple d'ARN de l'alias:

  `arn:aws:kms:us-east-2:123456789012:alias/alias-name`

- Exemple d'ID de clé :

  `12345678-1234-1234-1234-123456789012`

- Exemple de nom d'alias :

  `alias/MyAliasName`

Vous pouvez créer une clé gérée par le client à l'aide de la AWS Management Console ou de l'API AWS KMS. Les commandes de la AWS CLI suivantes créent une clé gérée par le client dans la Région AWS 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 AWS KMS key.

Linux & macOS

```
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.

Si vous ne spécifiez le paramètre `with-decryption`, ou si vous spécifiez le paramètre `no-with-decryption`, la commande renvoie un GUID chiffré.

Linux & macOS

```
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
```

4. Exécutez la commande suivante pour afficher les métadonnées du paramètre.

   Linux & macOS

   ```
   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 AWS KMS key gérée par le client.

   Linux & macOS

   ```
   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 AWS KMS key gérée par le client.

Linux & macOS

```
aws ssm put-parameter \
    --name "the-name-that-you-specified" \
    --value "a-new-parameter-value" \
    --type "SecureString" \
    --key-id "the-KMSkey-ID" \
    --overwrite
```

```
aws ssm put-parameter \
    --name "the-name-that-you-specified" \
    --value "a-new-parameter-value" \
    --type "SecureString" \
    --key-id "account-alias/the-KMSkey-ID" \
    --overwrite
```

Windows

```
aws ssm put-parameter ^
    --name "the-name-that-you-specified" ^
    --value "a-new-parameter-value" ^
    --type "SecureString" ^
    --key-id "the-KMSkey-ID" ^
    --overwrite
```

```
aws ssm put-parameter ^
    --name "the-name-that-you-specified" ^
    --value "a-new-parameter-value" ^
    --type "SecureString" ^
    --key-id "account-alias/the-KMSkey-ID" ^
    --overwrite
```

6. Exécutez la commande suivante pour afficher la dernière valeur du paramètre.

Linux & macOS

```
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 & macOS

```
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 sélectionnez `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'opération d'API Decrypt AWS KMS. Pour obtenir des informations sur la configuration du contrôle d'accès pour une AWS KMS, consultez Authentification et contrôle d'accès pour une AWS KMS dans le Manuel du développeur AWS Key Management Service.

> **Important**
>
> Si vous modifiez l'alias de clé KMS pour la clé KMS utilisée pour chiffrer un paramètre, vous devez alors également mettre à jour l'alias de clé utilisé par le paramètre pour référencer la AWS KMS. Cela s'applique uniquement à l'alias de clé KMS. L'ID de clé auquel un alias s'attache ne change pas, sauf si vous supprimez la clé entière.

Créer un paramètre multi-ligne (AWS CLI)

Vous pouvez utiliser la AWS CLI pour créer un paramètre avec des sauts de ligne. Utilisez des sauts de ligne pour diviser le texte en valeurs de paramètres plus longues afin d'améliorer la lisibilité ou, par exemple, mettre à jour le contenu de paramètres multi-paragraphes d'une page Web. Vous pouvez inclure le contenu dans un fichier JSON et utiliser l'option `--cli-input-json` en utilisant des caractères de saut de ligne tels que `\n`, comme illustré dans l'exemple suivant.

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS Command Line Interface (AWS CLI).

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l'AWS CLI.

2. Exécutez la commande suivante pour créer un paramètre multiligne.

   Linux & macOS

   ```
   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 de paramètre enregistrée est stockée de la façon suivante.

```
<para>Paragraph One</para>
<para>Paragraph Two</para>
<para>Paragraph Three</para>
```

Créer un paramètre Systems Manager (Tools for Windows PowerShell)

Vous pouvez utiliser AWS Tools for Windows PowerShell pour créer des types de paramètres `String`, `StringList` et `SecureString`. Après avoir supprimé un paramètre, attendez au moins 30 secondes avant de créer un paramètre avec le même nom.

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.

> **Note**
>
> Les paramètres sont uniquement disponibles dans la Région AWS où ils ont été créés.

Rubriques

- Créer un paramètre String (Tools for Windows PowerShell)
- Créer un paramètre StringList (Tools for Windows PowerShell)
- Créer un paramètre SecureString (Tools for Windows PowerShell)

Créer un paramètre **`String`** (Tools for Windows PowerShell)

1. Si vous ne l'avez pas déjà fait, installez et configurez AWS Tools for PowerShell (outils pour Windows PowerShell).

   Pour plus d'informations, consultez Installation d'AWS Tools for PowerShell.

2. Exécutez la commande suivante pour créer un paramètre contenant une valeur de texte brut. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

```
Write-SSMParameter `
    -Name "parameter-name" `
    -Value "parameter-value" `
    -Type "String"
```

-ou-

Exécutez la commande suivante pour créer un paramètre qui contient un ID d'Amazon Machine Image (AMI) comme valeur de paramètre.

> **Note**
>
> Pour créer un paramètre avec une balise, créez au préalable la service.model.tag en tant que variable. Voici un exemple.
>
> ```
> $tag = New-Object Amazon.SimpleSystemsManagement.Model.Tag
> $tag.Key = "tag-key"
> $tag.Value = "tag-value"
> ```

```
Write-SSMParameter `
    -Name "parameter-name" `
    -Value "an-AMI-id" `
    -Type "String" `
    -DataType "aws:ec2:image" `
    -Tags $tag
```

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 de plus amples informations, veuillez consulter Prise en charge de paramètres natifs pour les ID d'Amazon Machine Image.

Voici un exemple qui utilise une hiérarchie de paramètres.

```
Write-SSMParameter `
```

```
    -Name "/IAD/Web/SQL/IPaddress" `
    -Value "99.99.99.999" `
    -Type "String" `
    -Tags $tag
```

3. Exécutez la commande suivante pour vérifier les détails du paramètre.

```
(Get-SSMParameterValue -Name "the-parameter-name-you-specified").Parameters
```

Créer un paramètre **`StringList`** (Tools for Windows PowerShell)

1. Si vous ne l'avez pas déjà fait, installez et configurez AWS Tools for PowerShell (outils pour Windows PowerShell).

   Pour plus d'informations, consultez Installation d'AWS Tools for PowerShell.

2. Exécutez la commande suivante pour créer un paramètre StringList. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   > **Note**
   >
   > Pour créer un paramètre avec une balise, créez au préalable la service.model.tag en tant que variable. Voici un exemple.
   >
   > ```
   > $tag = New-Object Amazon.SimpleSystemsManagement.Model.Tag
   > $tag.Key = "tag-key"
   > $tag.Value = "tag-value"
   > ```

   ```
   Write-SSMParameter `
       -Name "parameter-name" `
       -Value "a-comma-separated-list-of-values" `
       -Type "StringList" `
       -Tags $tag
   ```

   En cas de réussite, la commande renvoie le numéro de version du paramètre.

   Voici un exemple.

```
Write-SSMParameter `
    -Name "stringlist-parameter" `
    -Value "Milana,Mariana,Mark,Miguel" `
    -Type "StringList" `
    -Tags $tag
```

> **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 (Tools for Windows PowerShell)

Avant de créer un paramètre `SecureString`, lisez les exigences liées à ce type de paramètre. Pour de plus amples informations, veuillez consulter Créer un paramètre SecureString (AWS CLI).

> **Important**
>
> Seule la valeur d'un paramètre `SecureString` est chiffrée. Les noms de paramètres, les descriptions et d'autres propriétés ne sont pas chiffrés.

> **Important**
>
> Parameter Store prend uniquement en charge des clés KMS à chiffrement symétrique. Vous ne pouvez pas utiliser une clé KMS à chiffrement asymétrique pour chiffrer vos paramètres. Pour savoir si une clés KMS est symétrique ou asymétrique, consultez Identification de clés symétriques et asymétriques dans le Guide du développeur AWS Key Management Service.

1. Si vous ne l'avez pas déjà fait, installez et configurez AWS Tools for PowerShell (outils pour Windows PowerShell).

   Pour plus d'informations, consultez Installation d'AWS Tools for PowerShell.

2. Exécutez la commande suivante pour créer un paramètre. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   > **Note**
   >
   > Pour créer un paramètre avec une balise, créez d'abord la service.model.tag en tant que variable. Voici un exemple.
   >
   > ```
   > $tag = New-Object Amazon.SimpleSystemsManagement.Model.Tag
   > $tag.Key = "tag-key"
   > $tag.Value = "tag-value"
   > ```

   ```
   Write-SSMParameter `
       -Name "parameter-name" `
       -Value "parameter-value" `
       -Type "SecureString"  `
       -KeyId "an AWS KMS key ID, an AWS KMS key ARN, an alias name, or an alias ARN"
    `
       -Tags $tag
   ```

   En cas de réussite, la commande renvoie le numéro de version du paramètre.

   > **Note**
   >
   > Pour utiliser la Clé 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 gérée par AWS.

   ```
   Write-SSMParameter `
       -Name "/Finance/Payroll/3l3vat3131" `
   ```

```
    -Value "P@sSwW)rd" `
    -Type "SecureString"`
    -Tags $tag
```

3. Exécutez la commande suivante pour vérifier les détails du paramètre.

```
(Get-SSMParameterValue -Name "the-parameter-name-you-specified" -WithDecryption
 $true).Parameters
```

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'opération d'API Decrypt AWS KMS. Pour obtenir des informations sur la configuration du contrôle d'accès pour une AWS KMS, consultez Authentification et contrôle d'accès pour une AWS KMS dans le Manuel du développeur AWS Key Management Service.

> **Important**
>
> Si vous modifiez l'alias de clé KMS pour la clé KMS utilisée pour chiffrer un paramètre, vous devez alors également mettre à jour l'alias de clé utilisé par le paramètre pour référencer la AWS KMS. Cela s'applique uniquement à l'alias de clé KMS. L'ID de clé auquel un alias s'attache ne change pas, sauf si vous supprimez la clé entière.

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. Vous pouvez utiliser la AWS Systems Manager console, le AWS Command Line Interface (AWS CLI) AWS Tools for PowerShell, le ou l'DescribeParametersAPI pour rechercher des paramètres.

Rubriques

- Rechercher un paramètre (console)
- Rechercher un paramètre (AWS CLI)

Rechercher un paramètre (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Parameter Store.
3. Cliquez dans la zone de recherche et sélectionnez le mode de recherche. Par exemple, `Type` ou `Name`.
4. Fournissez des informations sur le type de recherche que vous avez sélectionné. Par exemple :
   - Si vous effectuez une recherche par `Type`, sélectionnez parmi `String`, `StringList` ou `SecureString`.
   - Si vous effectuez une recherche par `Name`, `contains`, sélectionnez`equals` ou `begins-with`, puis entrez tout ou partie d'un nom de paramètre.

   > **Note**
   >
   > Dans la console, le type de recherche par défaut `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 les différentes options que vous pouvez utiliser pour afficher les informations relatives aux paramètres de votre Compte AWS. Pour plus d'informations sur ces options, consultez describe-parameters dans le Guide de l'utilisateur AWS Command Line Interface .

1. Installez et configurez le AWS Command Line Interface (AWS CLI), si ce n'est pas déjà fait.

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l' AWS CLI.
2. Remplacez les exemples de valeurs dans les commandes suivantes par des valeurs reflétant les paramètres créés dans votre compte.

Linux & macOS

```
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 équivalent 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 AWS KMS clé par défaut de votre compte. Pour plus d'informations, consultez Créer un paramètre SecureString (AWS CLI).

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": []
```

```
        }
    ]
}
```

Affectation de politiques de paramètres

Les politiques 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 politiques de paramètres sont particulièrement utiles pour vous obliger à mettre à jour ou à supprimer les mots de passe et les données de configuration stockés dansParameter Store, une fonctionnalité de AWS Systems Manager. Parameter Storepropose les types de politiques suivants : `ExpirationExpirationNotification`, et`NoChangeNotification`.

> **Note**
>
> Pour implémenter les cycles de vie de rotation des mots de passe, utilisez. AWS Secrets Manager Vous pouvez effectuer une rotation, gérer et récupérer les informations d'identification de la base de données, les clés d'API et d'autres secrets tout au long de leur cycle de vie à l'aide de Secrets Manager. Pour plus d'informations, voir Qu'est-ce que c'est AWS Secrets Manager ? dans le guide de AWS Secrets Manager l'utilisateur.

Parameter Store applique les politiques de paramètre à l'aide d'analyses asynchrones et périodiques. Une fois que vous avez créé une politique, vous n'avez pas besoin d'effectuer d'autres actions pour l'appliquer. Parameter Store exécute automatiquement l'action définie par la politique en fonction des critères que vous avez spécifiés.

> **Note**
>
> Les politiques de paramètre sont disponibles pour les paramètres qui utilisent le niveau de paramètres avancés. Pour plus d'informations, consultez Gestion des niveaux de paramètres.

Une politique de paramètre est une séquence JSON, comme illustré dans le tableau suivant. Vous pouvez attribuer une politique lorsque vous créez un nouveau paramètre avancé ; vous pouvez aussi appliquer une politique en mettant à jour un paramètre. Parameter Store prend en charge les types suivants de politiques de paramètre.

Politique	Détails	Exemples
Expiration	Cette politique 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 politique. La mise à jour d'un paramètre n'affecte pas la date ou l'heure d'expiration de la politique qui lui 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`.	`{ "Type": "Expiration", "Version": "1.0", "Attributes": { "Timestamp": "2018-12-02T21:34:33.000Z" } }`

Politique	Détails	Exemples
ExpirationNotification	Cette politique déclenche un événement sur Amazon EventBridge (EventBridge) qui vous informe de l'expiration. Cette politique 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 politique déclenche un événement EventBridge si un paramètre n'a pas été modifié pendant une période spécifiée . Ce type de politique est utile quand, par exemple, un mot de passe doit être modifié dans un délai donné. Cette politique 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 politiques à un paramètre. Par exemple, vous pouvez attribuer `Expiration` des `ExpirationNotification` politiques afin que le système déclenche un EventBridge événement pour vous informer de la suppression imminente d'un paramètre. Vous pouvez attribuer un maximum de dix (10) politiques à un paramètre.

L'exemple suivant montre la syntaxe d'une demande d'PutParameterAPI qui attribue quatre politiques à un nouveau `SecureString` paramètre nomméProdDB3.

```
{
    "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 politiques à un paramètre existant

Cette section contient des informations sur la façon d'ajouter des politiques à un paramètre existant à l'aide de la AWS Systems Manager console, du AWS Command Line Interface (AWS CLI) et AWS Tools for Windows PowerShell . Pour plus d'informations sur la création d'un nouveau paramètre incluant des politiques, consultez Création de paramètres Systems Manager.

Rubriques

- Ajouter des politiques à un paramètre existant (console)
- Ajouter des politiques à un paramètre existant (AWS CLI)
- Ajouter des politiques à un paramètre existant (Outils pour Windows PowerShell)

Ajouter des politiques à un paramètre existant (console)

Utilisez la procédure suivante pour ajouter des politiques à un paramètre en utilisant la console Systems Manager.

Pour ajouter des politiques à un paramètre existant

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Parameter Store.
3. Sélectionnez l'option en regard du paramètre que vous souhaitez mettre à jour afin d'inclure les politiques, puis sélectionnez Edit (Modifier).
4. Choisir Advanced (Avancé).
5. (Facultatif) Dans la section Parameter policies (Politiques de paramètre), sélectionnez Enabled (Activé). Vous pouvez spécifier une date d'expiration et une ou plusieurs politiques de notification pour ce paramètre.
6. Sélectionnez Enregistrer les modifications.

> **Important**
>
> - Parameter Store conserve les politiques associées à un paramètre jusqu'à ce que vous les remplaciez par de nouvelles politiques ou que vous les supprimiez.
> - Pour supprimer toutes les politiques d'un paramètre existant, modifiez le paramètre et appliquez une politique vide en utilisant des crochets et des accolades, comme suit : `[{}]`

- Si vous ajoutez une nouvelle politique à un paramètre qui en possède déjà, Systems Manager remplace les politiques déjà associées au paramètre. Les politiques existantes sont supprimées. Si vous souhaitez ajouter une nouvelle politique à un paramètre qui en possède déjà une ou plusieurs, vous devez copier et coller les politiques d'origine, saisir la nouvelle politique, puis enregistrer vos modifications.

Ajouter des politiques à un paramètre existant (AWS CLI)

Utilisez la procédure suivante pour ajouter des politiques à un paramètre existant à l'aide de l' AWS CLI.

Pour ajouter des politiques à un paramètre existant

1. Installez et configurez le AWS Command Line Interface (AWS CLI), si ce n'est pas déjà fait.

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l' AWS CLI.
2. Exécutez la commande suivante pour ajouter des politiques à un paramètre existant. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   Linux & macOS

   ```
   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 politique d'expiration qui supprime le paramètre au bout de 15 jours. L'exemple inclut également une politique de notification qui génère un EventBridge événement cinq (5) jours avant la suppression du paramètre. Enfin, il inclut une politique `NoChangeNotification` si aucune modification n'est apportée à ce paramètre au bout de 60 jours. L'exemple utilise un nom brouillé (`3l3vat3131`) comme paramètre de mot de passe et une AWS KMS key AWS Key Management Service . Pour plus d'informations AWS KMS keys, consultez la section AWS Key Management Service Concepts du guide du AWS Key Management Service développeur.

Linux & macOS

```
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. Remplacez *`parameter name`* (nom du paramètre) avec vos propres informations.

Linux & macOS

```
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 politiques d'un paramètre jusqu'à ce que vous les remplaciez par de nouvelles politiques ou que vous les supprimiez.
- Pour supprimer toutes les politiques d'un paramètre existant, modifiez le paramètre et appliquez une politique vide en l'entourant de crochets et d'accolades. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations. Par exemple :

  Linux & macOS

  ```
  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 politique à un paramètre qui en possède déjà, Systems Manager remplace les politiques déjà associées au paramètre. Les politiques existantes sont supprimées. Si vous souhaitez ajouter une nouvelle politique à un paramètre qui en

possède déjà une ou plusieurs, vous devez copier et coller les politiques d'origine, saisir la nouvelle politique, puis enregistrer vos modifications.

Ajouter des politiques à un paramètre existant (Outils pour Windows PowerShell)

Utilisez la procédure suivante pour ajouter des politiques à un paramètre existant à l'aide des Outils pour Windows PowerShell. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

Pour ajouter des politiques à un paramètre existant

1. Ouvrez Outils pour Windows PowerShell et exécutez la commande suivante pour spécifier vos informations d'identification. Vous devez soit disposer des autorisations d'administrateur dans Amazon Elastic Compute Cloud (Amazon EC2), soit avoir obtenu les autorisations appropriées AWS Identity and Access Management dans (IAM).

   ```
   Set-AWSCredentials `
       -AccessKey access-key-name `
       -SecretKey secret-key-name
   ```

2. Exécutez la commande suivante pour définir la région de votre PowerShell session. L'exemple utilise la région USA Est (Ohio) (us-east-2).

   ```
   Set-DefaultAWSRegion `
       -Region us-east-2
   ```

3. Exécutez la commande suivante pour ajouter des politiques à un paramètre existant. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   ```
   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 politique d'expiration qui supprime le paramètre à minuit (GMT) le 13 mai 2020. L'exemple inclut également une politique de notification qui génère un EventBridge

événement cinq (5) jours avant la suppression du paramètre. Enfin, il inclut une politique `NoChangeNotification` si aucune modification n'est apportée à ce paramètre au bout de 60 jours. L'exemple utilise un nom brouillé (`3l3vat3131`) comme paramètre de mot de passe et une Clé gérée par 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. Remplacez *parameter name* (nom du paramètre) avec vos propres informations.

```
(Get-SSMParameterValue -Name "parameter name").Parameters
```

Important

- Parameter Store conserve les politiques associées à un paramètre jusqu'à ce que vous les remplaciez par de nouvelles politiques ou que vous les supprimiez.
- Pour supprimer toutes les politiques d'un paramètre existant, modifiez le paramètre et appliquez une politique vide en l'entourant de crochets et d'accolades. Par exemple :

  ```
  Write-SSMParameter `
      -Name "parameter name" `
      -Value "parameter value" `
      -Type "parameter type" `
      -Policies "[{}]"
  ```

- Si vous ajoutez une nouvelle politique à un paramètre qui en possède déjà, Systems Manager remplace les politiques déjà associées au paramètre. Les politiques existantes sont supprimées. Si vous souhaitez ajouter une nouvelle politique à un paramètre qui en

possède déjà une ou plusieurs, vous devez copier et coller les politiques d'origine, saisir la nouvelle politique, puis enregistrer vos modifications.

Utiliser 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
- Interrogation de paramètres dans une hiérarchie
- Restriction de l'accès aux opérations de l'API Parameter Store
- Gérer les paramètres en utilisant les hiérarchies (AWS CLI)

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 paramètres racine. 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 capacité de demander tous les paramètres dans une hiérarchie en utilisant l'opération d'API GetParametersByPath. Pour exemple, si vous

exécutez la commande suivante à partir de l'AWS Command Line Interface (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 opérations de l'API Parameter Store

À l'aide des politiques AWS Identity and Access Management (IAM), vous pouvez autoriser ou restreindre l'accès des utilisateurs aux opérations et au contenu de l'API Parameter Store.

Dans l'exemple de politique suivant, les utilisateurs sont d'abord autorisés à accéder pour exécuter l'opération d'API `PutParameter` sur tous les paramètres du Compte AWS 123456789012 dans la 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'opération `PutParameter`. En d'autres termes, les utilisateurs auxquels cette politique 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 vous 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 Command Line Interface (AWS CLI).

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l'AWS CLI.
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 & macOS

   ```
   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 & macOS

```
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 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 & macOS

   ```
   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 & macOS

   ```
   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 & macOS

   ```
   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 & macOS

   ```
   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 & macOS

   ```
   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, une nouvelle version est AWS

Systems Manager automatiquement enregistrée et le numéro de version est incrémenté 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, disons que vous avez un paramètre appelé `/MyApp/DB/ConnectionString`. La valeur de ce paramètre est une chaîne de connexion à un serveur MySQL dans 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 de la version 2 vers la version 3 du paramètre.

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 consulter une liste de versions de paramètres spécifiques qui utilisent toutes le même libellé si vous interrogez votre ensemble de paramètres à l'aide de l'opération d'GetParametersByPathAPI, comme décrit plus loin dans cette section.

> **Note**
>
> Si vous exécutez une commande qui spécifie une version d'un paramètre qui n'existe pas, la commande échoue. Il ne revient pas à la valeur la plus récente ou à la valeur par défaut du paramètre.

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 du paramètre 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 paramètre. Vous devez attacher une étiquette à une version spécifique d'un paramètre.
- Si vous ne souhaitez plus utiliser une étiquette de paramètre, vous pouvez la déplacer vers une autre version d'un paramètre ou la supprimer.
- 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)

Cette section explique comment effectuer les tâches suivantes à l'aide de la console Systems Manager.

Créer une 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 la version d'un paramètre

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Parameter Store.
3. Sélectionnez le nom d'un paramètre pour ouvrir la page des détails de ce paramètre.
4. Sélectionnez l'onglet History (Historique).
5. Sélectionnez la version du paramètre pour laquelle vous souhaitez attacher une étiquette.
6. Sélectionnez Manage labels (Gérer des étiquettes).
7. Sélectionnez Add new label (Ajouter une nouvelle étiquette).
8. Dans la zone de texte, saisissez le nom de l'étiquette. Pour ajouter d'autres étiquettes, sélectionnez Add new label (Ajouter une nouvelle étiquette). Vous pouvez attacher un maximum de dix étiquettes.
9. Lorsque vous avez terminé, sélectionnez Enregistrer les modifications.

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 la version d'un paramètre

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Parameter Store.
3. Sélectionnez le nom d'un paramètre pour ouvrir la page des détails de ce paramètre.
4. Sélectionnez 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.

Déplacer une étiquette de paramètre (console)

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

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Parameter Store.
3. Sélectionnez le nom d'un paramètre pour ouvrir la page des détails de ce paramètre.
4. Sélectionnez l'onglet History (Historique).
5. Sélectionnez la version du paramètre pour laquelle vous souhaitez déplacer l'étiquette.
6. Sélectionnez Manage labels (Gérer des étiquettes).
7. Sélectionnez Add new label (Ajouter une nouvelle étiquette).
8. Dans la zone de texte, saisissez le nom de l'étiquette.
9. Lorsque vous avez terminé, sélectionnez Enregistrer les modifications.

Supprimer des étiquettes de paramètre (console)

La procédure suivante décrit la suppression d'une ou plusieurs étiquettes de paramètres avec la console Systems Manager.

Pour supprimer des étiquettes d'un paramètre

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Parameter Store.
3. Sélectionnez le nom d'un paramètre pour ouvrir la page des détails de ce paramètre.
4. Sélectionnez l'onglet History (Historique).
5. Sélectionnez la version du paramètre pour laquelle vous souhaitez supprimer les étiquettes.
6. Sélectionnez Manage labels (Gérer des étiquettes).
7. Sélectionnez Remove (Supprimer) en regard de chaque étiquette à supprimer.
8. Lorsque vous avez terminé, sélectionnez Enregistrer les modifications.

9. Confirmez l'exactitude de vos modifications, saisissez `Confirm` dans la zone de texte, puis sélectionnez Confirm (Confirmer).

Utilisation des étiquettes de paramètre (AWS CLI)

Cette section explique comment effectuer les tâches suivantes à l'aide de l' AWS Command Line Interface (AWS CLI).

- Créer une nouvelle étiquette de paramètre (AWS CLI)
- Afficher les étiquettes d'un paramètre (AWS CLI)
- Afficher la liste des paramètres auxquels une étiquette est affectée (AWS CLI)
- Déplacer une étiquette de paramètre (AWS CLI)
- Supprimer des étiquettes de paramètre (AWS CLI)

Créer une nouvelle étiquette de paramètre (AWS CLI)

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 étiquette de paramètre

1. Installez et configurez le AWS Command Line Interface (AWS CLI), si ce n'est pas déjà fait.

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l' AWS CLI.
2. Exécutez la commande suivante pour afficher la liste des paramètres auxquels vous êtes autorisé à attacher une étiquette.

   > **Note**
   >
   > Les paramètres ne sont disponibles que Région AWS là 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.

```
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, séparez 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 telles que les 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.

Afficher les étiquettes d'un paramètre (AWS CLI)

Vous pouvez utiliser l'opération GetParameterHistoryAPI pour afficher l'historique complet et toutes les étiquettes associées à un paramètre spécifique. Vous pouvez également utiliser l'opération GetParametersByPathAPI pour afficher la liste de tous les paramètres auxquels une étiquette spécifique est attribuée.

Pour afficher les libellés d'un paramètre à l'aide de l'opération GetParameterHistory 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 du paramètre dont vous voulez afficher les étiquettes.

2. Exécutez la commande suivante pour afficher toutes les versions du paramètre.

   ```
   aws ssm get-parameter-history --name parameter-name --with-decryption
   ```

   Le système retourne des informations telles que les suivantes.

   ```
   {
       "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 la liste des paramètres auxquels une étiquette est affectée (AWS CLI)

Vous pouvez utiliser l'opération GetParametersByPathAPI pour afficher la liste de tous les paramètres d'un chemin auxquels une étiquette spécifique a été attribuée.

Exécutez la commande suivante pour afficher la liste des paramètres dans un chemin d'accès auxquels une étiquette spécifique est attribuée. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

```
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 retourne des informations telles que les suivantes. 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"
        }
    ]
}
```

Déplacer une étiquette de paramètre (AWS CLI)

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. Remplacez *parameter name* (nom du paramètre) avec vos propres informations.

   ```
   aws ssm get-parameter-history \
       --name "parameter name"
   ```

   Notez les versions des paramètres vers lesquelles, ou à partir desquelles, vous voulez déplacer l'étiquette.

2. Exécutez la commande suivante pour affecter une étiquette existante à une autre version d'un paramètre. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   ```
   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.

Supprimer des étiquettes de paramètre (AWS CLI)

La procédure suivante décrit la suppression des étiquettes de paramètres à l'aide de la AWS CLI.

Pour supprimer une étiquette de paramètre

1. Exécutez la commande suivante pour afficher toutes les versions du paramètre. Remplacez *parameter name* (nom du paramètre) avec vos propres informations.

```
aws ssm get-parameter-history \
    --name "parameter name"
```

Le système retourne des informations telles que les suivantes.

```
{
    "Parameters": [
        {
            "Name": "foo",
            "DataType": "text",
            "LastModifiedDate": 1607380761.11,
            "Labels": [
                "l3",
                "l2"
            ],
            "Value": "test",
            "Version": 1,
            "LastModifiedUser": "arn:aws:iam::123456789012:user/test",
            "Policies": [],
            "Tier": "Standard",
            "Type": "String"
        },
        {
            "Name": "foo",
            "DataType": "text",
            "LastModifiedDate": 1607380763.11,
            "Labels": [
                "l1"
            ],
            "Value": "test",
            "Version": 2,
            "LastModifiedUser": "arn:aws:iam::123456789012:user/test",
            "Policies": [],
            "Tier": "Standard",
```

```
            "Type": "String"
        }
    ]
}
```

Notez la version du paramètre pour laquelle vous souhaitez supprimer une ou plusieurs étiquettes.

2. Exécutez la commande suivante pour supprimer de ce paramètre les étiquettes que vous avez choisies. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

```
aws ssm unlabel-parameter-version \
    --name parameter name \
    --parameter-version version \
    --labels label 1,label 2,label 3
```

Le système retourne des informations telles que les suivantes.

```
{
    "InvalidLabels": ["invalid"],
    "DeletedLabels" : ["Prod"]
 }
```

Utilisation des versions de paramètre

Chaque fois que vous modifiez la valeur d'un paramètre, Parameter Store, une des fonctionnalités de AWS Systems Manager , 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 augmente automatiquement le numéro de version d'une unité. 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 API et les documents SSM, par exemple : `ssm:MyParameter:3`. 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. Si vous spécifiez le numéro d'une version qui n'existe pas, le système renvoie une erreur au lieu de revenir à la version la plus récente ou à la version par défaut du paramètre.

Vous pouvez également utiliser les versions de paramètre pour savoir le nombre 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 conserver jusqu'à 100 versions d'un paramètre. Lorsque 100 versions d'un paramètre ont été créées, à chaque nouvelle création d'une version, la version la plus ancienne du paramètre est supprimée de l'historique pour faire place à la nouvelle version.

Cette règle s'applique sauf lorsqu'il existe déjà 100 versions de paramètres dans l'historique et qu'une étiquette est affectée 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. Cela vise à empêcher la suppression de versions de paramètres auxquelles des étiquettes critiques ont été affectées. 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 obtenir des informations sur le déplacement d'étiquettes de paramètres, consultez Déplacer une étiquette de paramètre (console) et Déplacer une étiquette de paramètre (AWS CLI).

Les procédures suivantes vous montrent comment modifier un paramètre, puis vérifier qu'une nouvelle version a été créée. Vous pouvez utiliser les commandes `get-parameter` et `get-parameters` pour afficher les versions de paramètres. Pour des exemples d'utilisation de ces commandes, consultez GetParameteret GetParametersdans le Guide de référence de l'AWS Systems Manager API

Rubriques

- Créer une nouvelle version d'un paramètre (console)
- Référence d'une version de paramètre

Créer une nouvelle version d'un paramètre (console)

Vous pouvez utiliser la console 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 AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Parameter Store.

3. Sélectionnez 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.
4. Sélectionnez Edit (Modifier).
5. Dans la boîte de dialogue Value (Valeur), saisissez une nouvelle valeur, puis sélectionnez Save changes (Enregistrer les modifications).
6. Sélectionnez 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, sélectionnez 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 `run-instances command` Amazon Elastic Compute Cloud (Amazon EC2) utilise la version 3 du paramètre `golden-ami`.

Linux & macOS

```
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
```

Windows

```
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**
>
> L'utilisation de `resolve` et d'une valeur de paramètre ne fonctionne qu'avec l'option `--image-id` et un paramètre qui contient une Amazon Machine Image (AMI) comme valeur. Pour plus d'informations, consultez Prise en charge de paramètres natifs pour les ID d'Amazon Machine Image.

Voici un exemple pour spécifier la version 2 d'un paramètre nommé `MyRunCommandParameter` dans un document SSM.

YAML

```
---
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}}"
```

JSON

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

Utilisation de paramètres partagés

Le partage de paramètres avancés simplifie la gestion des données de configuration dans un environnement multi-comptes. Vous pouvez stocker et gérer vos paramètres de manière centralisée et les partager avec d'autres personnes Comptes AWS qui ont besoin de les référencer.

Parameter Stores'intègre à AWS Resource Access Manager (AWS RAM) pour permettre un partage de paramètres avancé. AWS RAM est un service qui vous permet de partager des ressources avec d'autres personnes Comptes AWS ou par le biais de AWS Organizations.

Avec AWS RAM, vous partagez les ressources que vous possédez en créant un partage de ressources. Un partage de ressources indique les ressources à partager, les autorisations à accorder et les consommateurs avec lesquels partager. Les consommateurs peuvent inclure :

- Spécifique Comptes AWS à l'intérieur ou à l'extérieur de son organisation dans AWS Organizations
- Une unité organisationnelle au sein de son organisation dans AWS Organizations
- Toute son organisation en AWS Organizations

Pour plus d'informations AWS RAM, consultez le guide de AWS RAM l'utilisateur.

Cette rubrique explique comment partager les paramètres que vous possédez et comment utiliser les paramètres partagés avec vous.

Table des matières

- Partage d'un paramètre
- Arrêter de partager un paramètre partagé
- Identification des paramètres partagés
- Accès aux paramètres partagés
- Ensembles d'autorisations pour le partage de paramètres
- Débit maximal pour les paramètres partagés
- Tarification des paramètres partagés
- Accès entre comptes pour les comptes fermés Comptes AWS

Conditions préalables au partage des paramètres

Les conditions suivantes doivent être remplies avant de pouvoir partager des paramètres depuis votre compte :

- Pour partager un paramètre, vous devez le posséder dans votre Compte AWS. Vous ne pouvez pas partager un paramètre qui a été partagé avec vous.
- Pour partager un paramètre, il doit se trouver dans le niveau des paramètres avancés. Pour plus d'informations sur les niveaux de paramètres, consultezGestion des niveaux de paramètres. Pour plus d'informations sur la modification d'un paramètre standard existant en paramètre avancé, consultezRemplacement d'un paramètre standard par un paramètre avancé.
- Pour partager un `SecureString` paramètre, il doit être chiffré à l'aide d'une clé gérée par le client, et vous devez partager la clé séparément AWS Key Management Service. Clés gérées par AWS ne peut pas être partagé. Les paramètres chiffrés par défaut Clé gérée par AWS peuvent être mis à jour pour utiliser à la place une clé gérée par le client. Pour les définitions AWS KMS clés, voir AWS KMS les concepts dans le guide du AWS Key Management Service développeur.
- Pour partager un paramètre avec votre organisation ou une unité organisationnelle dans AWS Organizations, vous devez activer le partage avec AWS Organizations. Pour plus d'informations, consultez Activation du partage avec AWS Organizations dans le Guide de l'utilisateur AWS RAM .

Partage d'un paramètre

Pour partager un paramètre, vous devez l'ajouter à un partage de ressources. Un partage de ressources est une AWS RAM ressource qui vous permet de partager vos ressources entre elles Comptes AWS. Un partage de ressources spécifie les ressources à partager, ainsi que les consommateurs avec qui elles seront partagées.

Lorsque vous partagez un paramètre que vous possédez avec d'autres utilisateurs Comptes AWS, vous pouvez choisir entre deux autorisations AWS gérées à accorder aux consommateurs. Pour plus d'informations, consultez Ensembles d'autorisations pour le partage de paramètres.

Si vous faites partie d'une organisation AWS Organizations et que le partage au sein de votre organisation est activé, vous pouvez autoriser les consommateurs de votre organisation à accéder au paramètre partagé depuis la AWS RAM console. Dans le cas contraire, les consommateurs reçoivent une invitation à rejoindre le partage de ressources et ont accès au paramètre partagé après avoir accepté l'invitation.

Vous pouvez partager un paramètre dont vous êtes propriétaire à l'aide de la AWS RAM console ou du AWS CLI.

> **Note**
>
> Bien que vous puissiez partager un paramètre à l'aide de l'opération de l'API Systems Manager PutResourcePolicy, nous vous recommandons d'utiliser AWS Resource Access Manager (AWS RAM) à la place. En effet, l'utilisation `PutResourcePolicy` nécessite l'étape supplémentaire consistant à promouvoir le paramètre au niveau d'un partage de ressources standard à l'aide de l'opération AWS RAM PromoteResourceShareCreatedFromPolicyAPI. Dans le cas contraire, le paramètre ne sera pas renvoyé par l'opération d'DescribeParametersAPI Systems Manager à l'aide de l'`--share`option.

Pour partager un paramètre dont vous êtes propriétaire à l'aide de la AWS RAM console

Voir Création d'un partage de ressources AWS RAM dans le guide de AWS RAM l'utilisateur.

Effectuez les sélections suivantes au fur et à mesure que vous terminez la procédure :

- Sur la page Étape 1, pour Ressources, sélectionnez`Parameter Store Advanced Parameter`, puis cochez la case correspondant à chaque paramètre du niveau de paramètres avancés que vous souhaitez partager.
- Sur la page Étape 2, pour Autorisations gérées, choisissez l'autorisation à accorder aux consommateurs, comme décrit Ensembles d'autorisations pour le partage de paramètres plus loin dans cette rubrique.

Choisissez d'autres options en fonction de vos objectifs de partage de paramètres.

Pour partager un paramètre dont vous êtes propriétaire à l'aide du AWS CLI

Utilisez la create-resource-sharecommande pour ajouter des paramètres à un nouveau partage de ressources.

Utilisez la associate-resource-sharecommande pour ajouter des paramètres à un partage de ressources existant.

L'exemple suivant crée un nouveau partage de ressources pour partager des paramètres avec les consommateurs d'une organisation et d'un compte individuel.

```
aws ram create-resource-share \
    --name "MyParameter" \
    --resource-arns "arn:aws:ssm:us-east-2:123456789012:parameter/MyParameter" \
    --principals "arn:aws:organizations::123456789012:ou/o-63bEXAMPLE/ou-46xi-rEXAMPLE"
 "987654321098"
```

Arrêter de partager un paramètre partagé

Lorsque vous arrêtez de partager un paramètre partagé, le compte client ne peut plus accéder au paramètre.

Pour arrêter de partager un paramètre qui vous appartient, vous devez le supprimer du partage de ressources. Pour ce faire, vous pouvez utiliser la console Systems Manager, la console AWS RAM ou l' AWS CLI.

Pour arrêter de partager un paramètre dont vous êtes propriétaire à l'aide de la AWS RAM console

Voir Mettre à jour un partage de ressources AWS RAM dans le guide de AWS RAM l'utilisateur.

Pour arrêter de partager un paramètre dont vous êtes propriétaire à l'aide du AWS CLI

Utilisez la commande disassociate-resource-share.

Identification des paramètres partagés

Les propriétaires et les consommateurs peuvent identifier les paramètres partagés à l'aide du AWS CLI.

Pour identifier les paramètres partagés à l'aide du AWS CLI

Pour identifier les paramètres partagés à l'aide de AWS CLI, vous pouvez choisir entre la `describe-parameters` commande Systems Manager et la AWS RAM `list-resources` commande.

Lorsque vous utilisez l'`--shared`option with`describe-parameters`, la commande renvoie les paramètres partagés avec vous.

Voici un exemple :

```
aws ssm describe-parameters --shared
```

Accès aux paramètres partagés

Les consommateurs peuvent accéder aux paramètres partagés à l'aide des outils de ligne de AWS commande et AWS des SDK. Pour les comptes clients, les paramètres partagés avec ce compte ne sont pas inclus dans la page Mes paramètres.

Exemple de CLI : accès aux détails des paramètres partagés à l'aide du AWS CLI

Pour accéder aux détails des paramètres partagés à l'aide de AWS CLI, vous pouvez utiliser les get-parameterscommandes get-parameterou. Vous devez spécifier le paramètre ARN complet `--name` afin de récupérer le paramètre depuis un autre compte.

Voici un exemple.

```
aws ssm get-parameter \
    --name arn:aws:ssm:us-east-2:123456789012:parameter/MySharedParameter
```

Intégrations prises en charge et non prises en charge pour les paramètres partagés

Actuellement, vous pouvez utiliser des paramètres partagés dans les scénarios d'intégration suivants :

- AWS CloudFormation paramètres du modèle
- L'extension Lambda AWS Parameters and Secrets
- Modèles de lancement d'Amazon Elastic Compute Cloud (EC2)
- Valeurs à utiliser `ImageID` avec la RunInstances commande EC2 pour créer des instances à partir d'un Amazon Machine Image () AMI
- Récupération de valeurs de paramètres dans des runbooks pour Automation, une fonctionnalité de Systems Manager

Les scénarios et services intégrés suivants ne prennent actuellement pas en charge l'utilisation de paramètres partagés :

- Paramètres dans les commandesRun Command, une fonctionnalité de Systems Manager
- AWS CloudFormation références dynamiques
- Les valeurs des variables d'environnement dans AWS CodeBuild
- Les valeurs des variables d'environnement dans AWS App Runner
- La valeur d'un secret dans Amazon Elastic Container Service

Ensembles d'autorisations pour le partage de paramètres

Les comptes clients bénéficient d'un accès en lecture seule aux paramètres que vous partagez avec eux. Le consommateur ne peut ni mettre à jour ni supprimer le paramètre. Le consommateur ne peut pas partager le paramètre avec un troisième compte.

Lorsque vous créez un partage de ressources AWS Resource Access Manager pour partager vos paramètres, vous pouvez choisir parmi deux ensembles d'autorisations AWS gérées pour accorder cet accès en lecture seule :

AWSRAMDefaultPermissionSSMParameterReadOnly

Actions autorisées :`DescribeParameters`,`GetParameter`, `GetParameters`

AWSRAMPermissionSSMParameterReadOnlyWithHistory

Actions autorisées :`DescribeParameters`,`GetParameter`,`GetParameters`, `GetParameterHistory`

Lorsque vous suivez les étapes décrites dans la section Création d'un partage de ressources AWS RAM dans le Guide de AWS RAM l'utilisateur, choisissez `Parameter Store Advanced Parameters` le type de ressource et l'une de ces autorisations gérées, selon que vous souhaitez que les utilisateurs consultent ou non l'historique des paramètres.

Débit maximal pour les paramètres partagés

Systems Manager limite le débit maximal (transactions par seconde) pour les opérations GetParameteret GetParameters. Le débit est appliqué au niveau du compte individuel. Par conséquent, chaque compte consommant un paramètre partagé peut utiliser son débit maximal

autorisé sans être affecté par les autres comptes. Pour plus d'informations sur le débit maximal pour les paramètres, consultez les rubriques suivantes :

- Augmenter Parameter Store le débit
- Quotas du service Systems Manager dans le Référence générale d'Amazon Web Services.

Tarification des paramètres partagés

Le partage entre comptes n'est disponible que dans le niveau de paramètres avancés. Pour les paramètres avancés, des frais sont facturés au prix actuel pour le stockage et l'utilisation de l'API pour chaque paramètre avancé. Le compte propriétaire est débité pour le stockage du paramètre avancé. Tout compte consommateur qui effectue un appel d'API vers un paramètre avancé partagé est facturé pour l'utilisation du paramètre.

Par exemple, si le compte A crée un paramètre avancé`MyAdvancedParameter`, ce compte est débité de 0,05 USD par mois pour stocker le paramètre.

Le compte A est ensuite partagé `MyAdvancedParameter` avec le compte B et le compte C. Pendant un mois, les trois comptes passent des appels vers`MyAdvancedParameter`. Le tableau suivant indique les frais qu'ils encourraient pour le nombre d'appels que chacun effectue.

> **Note**
>
> Les frais indiqués dans le tableau suivant sont fournis à titre d'illustration uniquement. Pour vérifier les prix actuels, consultez la section AWS Systems Manager Tarification pour Parameter Store.

Compte	Nombre d'appels	Frais
Compte A (compte propriétaire)	10 000 appels	• Stockage avancé des paramètres pendant un mois : 0,05 USD • 10 000 appels vers `MyAdvancedParameter` : 0,05 USD • Total : 0,10 USD

Compte	Nombre d'appels	Frais
Compte B (compte consommateur)	20 000 appels	• 20 000 appels vers `MyAdvancedParameter` : 0,10 USD • Total : 0,10 USD
Compte C (compte consommateur)	30 000 appels	• 30 000 appels vers `MyAdvancedParameter` : 0,15 USD • Total : 0,15 USD

Accès entre comptes pour les comptes fermés Comptes AWS

Si le Compte AWS compte propriétaire d'un paramètre partagé est fermé, tous les comptes consommateurs perdent l'accès au paramètre partagé. Si le compte propriétaire est rouvert dans les 90 jours suivant sa fermeture, les comptes consommateurs retrouvent l'accès aux paramètres précédemment partagés. Pour plus d'informations sur la réouverture d'un compte après sa fermeture, consultez la section Accès à votre compte Compte AWS après sa fermeture dans le Guide de AWS Account Management référence.

Utiliser des paramètres avec des commandes Run Command

Vous pouvez travailler avec des paramètres dansRun Command, une fonctionnalité de AWS Systems Manager. Pour plus d'informations, consultez AWS Systems Manager Run Command.

Exécuter un paramètre String (console)

La procédure suivante vous guide à travers le processus d'exécution d'une commande utilisant un paramètre `String`.

Pour exécuter un paramètre de chaîne à l'aide de Parameter Store

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Run Command.
3. Sélectionnez Run Command (Exécuter la commande).

4. Dans la liste Command document (Document de commande), sélectionnez `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 Targets (Cibles), sélectionnez les nœuds gérés sur lesquels vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant des instances ou des appareils de périphérie manuellement ou en spécifiant un groupe de ressources.

   > **Tip**
   >
   > Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés 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. Pour Rate control (Contrôle de débit) :

   - Dans Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage de nœuds gérés sur lesquels exécuter simultanément la commande.

     > **Note**
     >
     > Si vous avez sélectionné des cibles en spécifiant les balises appliquées aux nœuds gérés ou en spécifiant des groupes de ressources AWS , et que vous n'êtes pas certain du nombre de nœuds gérés ciblés, limitez le nombre de cibles autorisées à exécuter simultanément le document en indiquant un pourcentage.

   - Dans Error threshold (Seuil d'erreur), indiquez quand arrêter l'exécution de la commande sur les autres nœuds gérés après l'échec de celle-ci sur un certain nombre ou un certain pourcentage de nœuds. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les nœuds gérés sur lesquels la commande est toujours 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). Saisissez 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 (pour les instances EC2) ou de la fonction du service IAM (pour les machines activées par un système hybride) attribués à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, assurez-vous que le profil d'instance ou le rôle de service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

10. Dans la section SNS notifications (Notifications SNS), si vous souhaitez envoyer des notifications sur le statut d'exécution des commandes, cochez la case Enable SNS notifications (Activer les notifications SNS).

    Pour plus d'informations sur la configuration des notifications Amazon SNS pour Run Command, consultez Surveillance des changements d'état du Systems Manager à l'aide des notifications Amazon SNS.

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'un nœud dans lequel vous avez exécuté la commande, puis sélectionnez 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`**.

Exécuter un paramètre (AWS CLI)

Exemple 1 : commande simple

L'exemple de commande suivant inclut un paramètre Systems Manager appelé `DNS-IP`. La valeur de ce paramètre est simplement l'adresse IP d'un nœud. Cet exemple utilise une commande AWS Command Line Interface (AWS CLI) pour faire écho à la valeur du paramètre.

Linux & macOS

```
aws ssm send-command \
    --document-name "AWS-RunShellScript" \
    --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
```

La commande renvoie des informations telles que les suivantes.

```
{
    "Command": {
        "CommandId": "c70a4671-8098-42da-b885-89716EXAMPLE",
        "DocumentName": "AWS-RunShellScript",
        "DocumentVersion": "1",
        "Comment": "",
        "ExpiresAfter": "2023-12-26T15:19:17.771000-05:00",
        "Parameters": {
            "commands": [
                "echo {{ssm:DNS-IP}}"
            ]
```

```
        },
        "InstanceIds": [],
        "Targets": [
            {
                "Key": "instanceids",
                "Values": [
                    "i-02573cafcfEXAMPLE"
                ]
            }
        ],
        "RequestedDateTime": "2023-12-26T14:09:17.771000-05:00",
        "Status": "Pending",
        "StatusDetails": "Pending",
        "OutputS3Region": "us-east-2",
        "OutputS3BucketName": "",
        "OutputS3KeyPrefix": "",
        "MaxConcurrency": "50",
        "MaxErrors": "0",
        "TargetCount": 0,
        "CompletedCount": 0,
        "ErrorCount": 0,
        "DeliveryTimedOutCount": 0,
        "ServiceRole": "",
        "NotificationConfig": {
            "NotificationArn": "",
            "NotificationEvents": [],
            "NotificationType": ""
        },
        "CloudWatchOutputConfig": {
            "CloudWatchLogGroupName": "",
            "CloudWatchOutputEnabled": false
        },
        "TimeoutSeconds": 600,
        "AlarmConfiguration": {
            "IgnorePollAlarmFailure": false,
            "Alarms": []
        },
        "TriggeredAlarms": []
    }
}
```


Une fois l'exécution d'une commande terminée, vous pouvez afficher plus d'informations à son sujet à l'aide des commandes suivantes :

- get-command-invocation : affiche des informations détaillées sur l'exécution de la commande.
- list-command-invocations : affiche l'état d'exécution des commandes sur un nœud géré spécifique.
- list-commands : affiche l'état d'exécution des commandes sur les nœuds gérés.

Exemple 2 : déchiffrer une valeur de paramètre **`SecureString`**

L'exemple de commande suivant utilise un `SecureString` paramètre nommé SecurePassword. La commande utilisée dans le champ `parameters` récupère et déchiffre la valeur du paramètre `SecureString`, puis réinitialise le mot de passe administrateur local sans avoir à le transmettre en texte clair.

Linux

```
aws ssm send-command \
        --document-name "AWS-RunShellScript" \
        --document-version "1" \
        --targets "Key=instanceids,Values=i-02573cafcfEXAMPLE" \
        --parameters '{"commands":["secure=$(aws ssm get-parameters --names
 SecurePassword --with-decryption --query Parameters[0].Value --output text --region
 us-east-2)","echo $secure | passwd myuser --stdin"]}' \
        --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
```

Exemple 3 : référencer un paramètre dans un document SSM

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 pas en charge les références aux paramètres `SecureString`. Cela signifie que pour utiliser les paramètres `SecureString` avec, par

exemple, la fonctionnalité Run Command, vous devez récupérer la valeur du paramètre avant de la passer à Run Command, comme illustré dans les exemples suivants.

Linux & macOS

```
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
```

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
```

Prise en charge de paramètres natifs pour les ID d'Amazon Machine Image

Lorsque vous créez un paramètre `String`, vous pouvez spécifier le type de données `aws:ec2:image`, afin de vous assurer que la valeur du paramètre que vous saisissez est un format d'ID d'Amazon Machine Image (AMI) 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 créer de nouvelles instances. Vous référencez ensuite ce paramètre dans vos modèles, commandes et scripts.

Par exemple, vous définissez le paramètre qui contient votre IDAMI préféré lorsque vous exécutez la commande `run-instances` Amazon Elastic Compute Cloud (Amazon EC2).

> **Note**
>
> L'utilisateur qui exécute cette commande doit disposer d'autorisations AWS Identity and Access Management (IAM) incluant l'opération d'`ssm:GetParameters`API pour que la valeur du paramètre soit validée. Sinon, le processus de création du paramètre échoue.

Linux & macOS

```
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
```

Windows

```
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 votre AMI préférée lorsque vous créez une instance à l'aide de la console Amazon EC2. Pour plus d'informations, consultez la section Utilisation d'un paramètre Systems Manager pour en trouver un AMI dans le guide de l'utilisateur Amazon EC2.

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.

Octroyer des autorisations pour créer un paramètre du type de données **aws:ec2:image**

À l'aide de politiques AWS Identity and Access Management (IAM), vous pouvez fournir ou restreindre l'accès des utilisateurs aux opérations et au contenu de Parameter Store l'API.

Pour créer un paramètre de type de `aws:ec2:image` données, l'utilisateur doit disposer à la fois des `ec2:DescribeImages` autorisations `ssm:PutParameter` et des autorisations.

L'exemple de politique suivant octroie aux utilisateurs l'autorisation d'appeler l'opération d'API `PutParameter` pour `aws:ec2:image`. Cela signifie que l'utilisateur peut ajouter un paramètre du type de données `aws:ec2:image` au système.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ssm:PutParameter",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "ec2:DescribeImages",
            "Resource": "*"
        }
    ]
}
```

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.

Un numéro de version de paramètre pourrait être généré avant la fin de l'opération de validation. L'opération peut ne pas être terminée même si un numéro de version de paramètre est généré.

Pour vérifier si vos paramètres ont été créés correctement, nous vous recommandons d'utiliser Amazon EventBridge pour vous envoyer des notifications concernant vos opérations `create` et celles relatives aux `update` 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 sur l'abonnement à Parameter Store des événements dans EventBridge, voirConfiguration de notifications ou d'actions de déclenchement basées sur des événements Parameter Store.

Suppression de paramètres Systems Manager

Cette rubrique décrit comment supprimer des paramètres que vous avez créés dansParameter Store, une fonctionnalité de AWS Systems Manager.

Pour supprimer un paramètre (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Parameter Store.
3. Sous l'onglet My parameters (Mes paramètres), activez la case à cocher en regard de chaque paramètre à supprimer.
4. Sélectionnez Delete (Supprimer).

5. Dans la boîte de dialogue de confirmation, sélectionnez Delete parameters (Supprimer les paramètres).

Pour supprimer un paramètre (AWS CLI)

- Exécutez la commande suivante :

  ```
  aws ssm delete-parameter --name "my-parameter"
  ```

  Remplacez *my-parameter* par le nom du paramètre à supprimer.

  Pour plus d'informations sur toutes les options disponibles pour une utilisation avec la `delete-parameter` commande, reportez-vous delete-parameterà la AWS Systems Manager section de la référence des AWS CLI commandes.

Utilisation de paramètres publics

Certains Services AWS publient des informations sur les artefacts courants sous forme de paramètres AWS Systems Manager publics. Par exemple, le service Amazon Elastic Compute Cloud (Amazon EC2) publie des informations sur des Amazon Machine Images (AMIs) sous forme de paramètres publics.

Sujets abordés dans ce guide

- Recherche de paramètres publics
- Appel de paramètres publics d'AMI
- Appel d'un paramètre public d'AMI optimisée pour ECS
- Appel d'un paramètre public d'AMI optimisée pour EKS
- Appel de paramètres publics pour les régions Services AWS, les points de terminaison, les zones de disponibilité, les zones locales et les zones de longueur d'onde

Articles de AWS blog connexes

- Requête pour Régions AWS, points de terminaison, etc. à l'aide de AWS Systems ManagerParameter Store
- Query for the latest Amazon Linux AMI IDs using AWS Systems ManagerParameter Store
- Query for the Latest Windows AMI Using AWS Systems ManagerParameter Store

Recherche de paramètres publics

Vous pouvez rechercher des paramètres publics an utilisant la console Parameter Store ou la AWS Command Line Interface.

Le nom d'un paramètre public commence par `aws/service/list`. La section suivante du nom correspond au service auquel appartient ce paramètre.

Voici une liste de certains services qui fournissent des paramètres publics :

- `ami-amazon-linux-latest`
- `ami-windows-latest`
- `appmesh`
- `aws-for-fluent-bit`
- `bottlerocket`
- `canonical`
- `cloud9`
- `datasync`
- `debian`
- `ecs`
- `eks`
- `freebsd`
- `global-infrastructure`
- `marketplace`
- `storagegateway`

Les paramètres publics ne sont pas tous publiés pour tous Région AWS.

Recherche des paramètres publics en utilisant la console Parameter Store

Vous devez avoir au moins un paramètre dans votre Compte AWS et Région AWS avant de pouvoir rechercher des paramètres publics à l'aide de la console.

Pour rechercher des paramètres publics en utilisant la console

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Parameter Store.
3. Sélectionnez l'onglet Public parameters (Paramètres publics).
4. Sélectionnez le menu déroulant Select a service (Sélectionner un service) Sélectionnez le service dont vous voulez utiliser les paramètres.
5. (Facultatif) Filtrez les paramètres appartenant au service que vous avez sélectionné en saisissant plus d'informations dans la barre de recherche.
6. Sélectionnez le paramètre public à utiliser.

Recherche de paramètres publics à l'aide du AWS CLI

Utilisez `describe-parameters` pour découvrir des paramètres publics.

Utilisez `get-parameters-by-path` pour obtenir le chemin réel d'un service répertorié dans `/aws/service/list`. Pour obtenir le chemin du service, supprimez `/list` du chemin. Par exemple, `/aws/service/list/ecs` devient `/aws/service/ecs`.

Pour récupérer une liste de paramètres publics appartenant à différents services dans Parameter Store, exécutez la commande suivante.

```
aws ssm get-parameters-by-path --path /aws/service/list
```

La commande renvoie des informations telles que les suivantes. Cet exemple a été tronqué faute d'espace.

```
{
    "Parameters": [
        {
            "Name": "/aws/service/list/ami-al-latest",
            "Type": "String",
            "Value": "/aws/service/ami-al-latest/",
            "Version": 1,
            "LastModifiedDate": "2021-01-29T10:25:10.902000-08:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/list/ami-al-latest",
            "DataType": "text"
```

```
        },
        {
            "Name": "/aws/service/list/ami-windows-latest",
            "Type": "String",
            "Value": "/aws/service/ami-windows-latest/",
            "Version": 1,
            "LastModifiedDate": "2021-01-29T10:25:12.567000-08:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/list/ami-windows-
latest",
            "DataType": "text"
        },
        {
            "Name": "/aws/service/list/aws-storage-gateway-latest",
            "Type": "String",
            "Value": "/aws/service/aws-storage-gateway-latest/",
            "Version": 1,
            "LastModifiedDate": "2021-01-29T10:25:09.903000-08:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/list/aws-storage-
gateway-latest",
            "DataType": "text"
        },
        {
            "Name": "/aws/service/list/global-infrastructure",
            "Type": "String",
            "Value": "/aws/service/global-infrastructure/",
            "Version": 1,
            "LastModifiedDate": "2021-01-29T10:25:11.901000-08:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/list/global-
infrastructure",
            "DataType": "text"
        }
    ]
}
```

Pour afficher les paramètres appartenant à un service spécifique, sélectionnez le service dans la liste générée après l'exécution de la commande précédente. Ensuite, faites un appel à `get-parameters-by-path` en utilisant le nom du service souhaité.

Par exemple, `/aws/service/global-infrastructure`. Le chemin peut être à un seul niveau (seuls les paramètres qui correspondent aux valeurs exactes données sont appelés) ou récursif (contient des éléments dans le chemin au-delà de ce que vous avez donné).

> **Note**
>
> Le `/aws/service/global-infrastructure` chemin n'est pas pris en charge pour les requêtes dans toutes les régions. Pour plus d'informations, veuillez consulter Appel de paramètres publics pour les régions Services AWS, les points de terminaison, les zones de disponibilité, les zones locales et les zones de longueur d'onde.

Si aucun résultat n'est obtenu pour le service que vous spécifiez, ajoutez l'indicateur `--recursive` et exécutez à nouveau la commande.

```
aws ssm get-parameters-by-path --path /aws/service/global-infrastructure
```

Cela renvoie tous les paramètres appartenant à `global-infrastructure`. Voici un exemple.

```
{
    "Parameters": [
        {
            "Name": "/aws/service/global-infrastructure/current-region",
            "Type": "String",
            "LastModifiedDate": "2019-06-21T05:15:34.252000-07:00",
            "Version": 1,
            "Tier": "Standard",
            "Policies": [],
            "DataType": "text"
        },
        {
            "Name": "/aws/service/global-infrastructure/version",
            "Type": "String",
            "LastModifiedDate": "2019-02-04T06:59:32.875000-08:00",
            "Version": 1,
            "Tier": "Standard",
            "Policies": [],
            "DataType": "text"
        }
    ]
}
```

Vous pouvez aussi afficher les paramètres appartenant à un service spécifique en utilisant le filtre `Option:BeginsWith`.

```
aws ssm describe-parameters --parameter-filters "Key=Name, Option=BeginsWith, Values=/
aws/service/ami-amazon-linux-latest"
```

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",
            "LastModifiedDate": "2021-01-26T13:39:40.686000-08:00",
            "Version": 25,
            "Tier": "Standard",
            "Policies": [],
            "DataType": "text"
        },
        {
            "Name": "/aws/service/ami-amazon-linux-latest/amzn-ami-hvm-x86_64-gp2",
            "Type": "String",
            "LastModifiedDate": "2021-01-26T13:39:40.807000-08:00",
            "Version": 25,
            "Tier": "Standard",
            "Policies": [],
            "DataType": "text"
        },
        {
            "Name": "/aws/service/ami-amazon-linux-latest/amzn-ami-hvm-x86_64-s3",
            "Type": "String",
            "LastModifiedDate": "2021-01-26T13:39:40.920000-08:00",
            "Version": 25,
            "Tier": "Standard",
            "Policies": [],
            "DataType": "text"
        }
    ]
}
```

> **Note**
>
> Les paramètres renvoyés peuvent être différents si vous utilisez `Option=BeginsWith` car le modèle de recherche utilisé est différent.

Appel de paramètres publics d'AMI

Les paramètres publics d'Amazon Elastic Compute Cloud (Amazon Amazon Machine Image EC2) AMI () sont disponibles pour Amazon Linux 1, Amazon Linux 2, Amazon Linux 2023 (AL2023) Windows Server et depuis les chemins suivants :

- Amazon Linux 1, Amazon Linux 2 et Amazon Linux 2023 : `/aws/service/ami-amazon-linux-latest`
- Windows Server: `/aws/service/ami-windows-latest`

Appel de paramètres AMI publics pour Amazon Linux 1, Amazon Linux 2 et Amazon Linux 2023

Vous pouvez consulter la liste de tous les Amazon Linux 1, Amazon Linux 2 et Amazon Linux 2023 (AL2023) actuels AMIs en Région AWS utilisant la commande suivante dans le AWS Command Line Interface (AWS CLI).

Linux & macOS

```
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/al2023-ami-kernel-6.1-arm64",
```

```
    "/aws/service/ami-amazon-linux-latest/al2023-ami-kernel-6.1-x86_64",
    "/aws/service/ami-amazon-linux-latest/al2023-ami-minimal-kernel-6.1-arm64",
    "/aws/service/ami-amazon-linux-latest/al2023-ami-minimal-kernel-6.1-x86_64",
    "/aws/service/ami-amazon-linux-latest/al2023-ami-minimal-kernel-default-arm64",
    "/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/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-kernel-5.10-hvm-x86_64-ebs",
    "/aws/service/ami-amazon-linux-latest/al2023-ami-kernel-default-arm64",
    "/aws/service/ami-amazon-linux-latest/al2023-ami-kernel-default-x86_64",
    "/aws/service/ami-amazon-linux-latest/al2023-ami-minimal-kernel-default-x86_64",
    "/aws/service/ami-amazon-linux-latest/amzn-ami-hvm-x86_64-ebs",
    "/aws/service/ami-amazon-linux-latest/amzn-ami-minimal-hvm-x86_64-ebs",
    "/aws/service/ami-amazon-linux-latest/amzn-ami-minimal-hvm-x86_64-s3",
    "/aws/service/ami-amazon-linux-latest/amzn2-ami-hvm-arm64-gp2",
    "/aws/service/ami-amazon-linux-latest/amzn2-ami-kernel-5.10-hvm-arm64-gp2",
    "/aws/service/ami-amazon-linux-latest/amzn2-ami-kernel-5.10-hvm-x86_64-gp2",
    "/aws/service/ami-amazon-linux-latest/amzn2-ami-minimal-hvm-arm64-ebs",
    "/aws/service/ami-amazon-linux-latest/amzn2-ami-minimal-hvm-x86_64-ebs"
]
```

Vous pouvez afficher les détails relatifs à ces AMIs, en particulier les ID et les Amazon Resource Names (ARN) des AMI, à l'aide de la commande suivante.

Linux & macOS

```
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 Région AWS prise en charge par AWS Systems Manager, par exemple `us-east-2` pour la région USA Est (Ohio). Pour obtenir une liste des valeurs *region* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la 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/al2023-ami-kernel-6.1-arm64",
            "Type": "String",
            "Value": "ami-0b1b8b24a6c8e5d8b",
            "Version": 69,
            "LastModifiedDate": "2024-03-13T14:05:09.583000-04:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/ami-amazon-linux-
latest/al2023-ami-kernel-6.1-arm64",
            "DataType": "text"
        },
        {
            "Name": "/aws/service/ami-amazon-linux-latest/al2023-ami-kernel-6.1-
x86_64",
            "Type": "String",
            "Value": "ami-0e0bf53f6def86294",
            "Version": 69,
            "LastModifiedDate": "2024-03-13T14:05:09.890000-04:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/ami-amazon-linux-
latest/al2023-ami-kernel-6.1-x86_64",
            "DataType": "text"
        },
        {
            "Name": "/aws/service/ami-amazon-linux-latest/al2023-ami-minimal-
kernel-6.1-arm64",
            "Type": "String",
            "Value": "ami-09951bb66f9e5b5a5",
            "Version": 69,
            "LastModifiedDate": "2024-03-13T14:05:10.197000-04:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/ami-amazon-linux-
latest/al2023-ami-minimal-kernel-6.1-arm64",
            "DataType": "text"
        }
    ]
}
```


Vous pouvez afficher les détails d'un élément spécifique en AMI utilisant l'opération d'GetParametersAPI avec le AMI nom complet, y compris le chemin. Voici un exemple de commande.

Linux & macOS

```
aws ssm get-parameters \
    --names /aws/service/ami-amazon-linux-latest/al2023-ami-kernel-6.1-arm64 \
    --region us-east-2
```

Windows

```
aws ssm get-parameters ^
    --names /aws/service/ami-amazon-linux-latest/al2023-ami-kernel-6.1-arm64 ^
    --region us-east-2
```

La commande renvoie les informations suivantes.

```
{
    "Parameters": [
        {
            "Name": "/aws/service/ami-amazon-linux-latest/al2023-ami-kernel-6.1-arm64",
            "Type": "String",
            "Value": "ami-0b1b8b24a6c8e5d8b",
            "Version": 69,
            "LastModifiedDate": "2024-03-13T14:05:09.583000-04:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/ami-amazon-linux-
latest/al2023-ami-kernel-6.1-arm64",
            "DataType": "text"
        }
    ],
    "InvalidParameters": []
}
```

Appel de paramètres publics d'AMI pour Windows Server

Vous pouvez afficher une liste de tous les éléments actuels Windows Server AMIs à Région AWS l'aide de la commande suivante dans le AWS CLI.

Linux & macOS

```
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/EC2LaunchV2-Windows_Server-2016-English-Full-
Base",
    "/aws/service/ami-windows-latest/Windows_Server-2016-English-Full-
SQL_2014_SP3_Enterprise",
    "/aws/service/ami-windows-latest/Windows_Server-2016-German-Full-Base",
    "/aws/service/ami-windows-latest/Windows_Server-2016-Japanese-Full-
SQL_2016_SP3_Standard",
    "/aws/service/ami-windows-latest/Windows_Server-2016-Japanese-Full-SQL_2017_Web",
    "/aws/service/ami-windows-latest/Windows_Server-2019-English-Core-
EKS_Optimized-1.25",
    "/aws/service/ami-windows-latest/Windows_Server-2019-Italian-Full-Base",
    "/aws/service/ami-windows-latest/Windows_Server-2022-Japanese-Full-
SQL_2019_Enterprise",
    "/aws/service/ami-windows-latest/Windows_Server-2022-Portuguese_Brazil-Full-Base",
    "/aws/service/ami-windows-latest/amzn2-ami-hvm-2.0.20191217.0-x86_64-gp2-mono",
    "/aws/service/ami-windows-latest/Windows_Server-2016-English-Deep-Learning",
    "/aws/service/ami-windows-latest/Windows_Server-2016-Japanese-Full-
SQL_2016_SP3_Web",
    "/aws/service/ami-windows-latest/Windows_Server-2016-Korean-Full-Base",
    "/aws/service/ami-windows-latest/Windows_Server-2019-English-STIG-Core",
    "/aws/service/ami-windows-latest/Windows_Server-2019-French-Full-Base",
    "/aws/service/ami-windows-latest/Windows_Server-2019-Japanese-Full-
SQL_2017_Enterprise",
    "/aws/service/ami-windows-latest/Windows_Server-2019-Korean-Full-Base",
    "/aws/service/ami-windows-latest/Windows_Server-2022-English-Full-SQL_2022_Web",
    "/aws/service/ami-windows-latest/Windows_Server-2022-Italian-Full-Base",
    "/aws/service/ami-windows-latest/amzn2-x86_64-SQL_2019_Express",
    "/aws/service/ami-windows-latest/EC2LaunchV2-Windows_Server-2016-English-Core-
Base",
    "/aws/service/ami-windows-latest/Windows_Server-2016-English-Full-
SQL_2019_Enterprise",
```

```
    "/aws/service/ami-windows-latest/Windows_Server-2016-English-Full-
SQL_2019_Standard",
    "/aws/service/ami-windows-latest/Windows_Server-2016-Portuguese_Portugal-Full-
Base",
    "/aws/service/ami-windows-latest/Windows_Server-2019-English-Core-
EKS_Optimized-1.24",
    "/aws/service/ami-windows-latest/Windows_Server-2019-English-Deep-Learning",
    "/aws/service/ami-windows-latest/Windows_Server-2019-English-Full-SQL_2017_Web",
    "/aws/service/ami-windows-latest/Windows_Server-2019-Hungarian-Full-Base
]
```

Vous pouvez afficher les détails relatifs à ces AMIs, en particulier les ID et les Amazon Resource Names (ARN) des AMI, à l'aide de la commande suivante.

Linux & macOS

```
aws ssm get-parameters-by-path \
    --path "/aws/service/ami-windows-latest" \
    --region region
```

Windows

```
aws ssm get-parameters-by-path ^
    --path "/aws/service/ami-windows-latest" ^
    --region region
```

région représente l'identifiant d'une région Région AWS prise en charge par AWS Systems Manager, par exemple `us-east-2` pour la région USA Est (Ohio). Pour obtenir une liste des valeurs *region* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la 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-windows-latest/EC2LaunchV2-Windows_Server-2016-
English-Full-Base",
            "Type": "String",
```

```
            "Value": "ami-0a30b2e65863e2d16",
            "Version": 36,
            "LastModifiedDate": "2024-03-15T15:58:37.976000-04:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/ami-windows-latest/
EC2LaunchV2-Windows_Server-2016-English-Full-Base",
            "DataType": "text"
        },
        {
            "Name": "/aws/service/ami-windows-latest/Windows_Server-2016-English-Full-
SQL_2014_SP3_Enterprise",
            "Type": "String",
            "Value": "ami-001f20c053dd120ce",
            "Version": 69,
            "LastModifiedDate": "2024-03-15T15:53:58.905000-04:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/ami-windows-latest/
Windows_Server-2016-English-Full-SQL_2014_SP3_Enterprise",
            "DataType": "text"
        },
        {
            "Name": "/aws/service/ami-windows-latest/Windows_Server-2016-German-Full-
Base",
            "Type": "String",
            "Value": "ami-063be4935453e94e9",
            "Version": 102,
            "LastModifiedDate": "2024-03-15T15:51:12.003000-04:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/ami-windows-latest/
Windows_Server-2016-German-Full-Base",
            "DataType": "text"
        }
    ]
}
```

Vous pouvez afficher les détails d'un élément spécifique en AMI utilisant l'opération d'GetParametersAPI avec le AMI nom complet, y compris le chemin. Voici un exemple de commande.

Linux & macOS

```
aws ssm get-parameters \
    --names /aws/service/ami-windows-latest/EC2LaunchV2-Windows_Server-2016-English-
Full-Base \
    --region us-east-2
```

Windows

```
aws ssm get-parameters ^
    --names /aws/service/ami-windows-latest/EC2LaunchV2-Windows_Server-2016-English-
Full-Base ^
    --region us-east-2
```

La commande renvoie les informations suivantes.

```
{
    "Parameters": [
        {
            "Name": "/aws/service/ami-windows-latest/EC2LaunchV2-Windows_Server-2016-
English-Full-Base",
            "Type": "String",
            "Value": "ami-0a30b2e65863e2d16",
            "Version": 36,
            "LastModifiedDate": "2024-03-15T15:58:37.976000-04:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/ami-windows-latest/
EC2LaunchV2-Windows_Server-2016-English-Full-Base",
            "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 Amazon Machine Images (AMIs) optimisée Amazon ECS en tant que paramètres publics. Les utilisateurs sont invités à utiliser cette AMI lors de la création d'un nouveau cluster Amazon Elastic Compute Cloud (Amazon EC2) pour Amazon ECS, car l'AMIs 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 Amazon ECS pour Amazon Linux 2. Pour consulter les commandes pour d'autres systèmes d'exploitation, veuillez consulter Récupération de métadonnées d'AMI optimisées Amazon ECS dans le Manuel du développeur Amazon Elastic Container Service.

Linux & macOS

```
aws ssm get-parameters \
    --names /aws/service/ecs/optimized-ami/amazon-linux-2/recommended
```

Windows

```
aws ssm get-parameters ^
    --names /aws/service/ecs/optimized-ami/amazon-linux-2/recommended
```

La commande renvoie des informations telles que les suivantes.

```
{
    "Parameters": [
        {
            "Name": "/aws/service/ecs/optimized-ami/amazon-linux-2/recommended",
            "Type": "String",
            "Value": "{\"schema_version\":1,\"image_name\":\"amzn2-ami-ecs-
hvm-2.0.20210929-x86_64-ebs\",\"image_id\":\"ami-0c38a2329ed4dae9a\",\"os\":\"Amazon
 Linux 2\",\"ecs_runtime_version\":\"Docker version 20.10.7\",\"ecs_agent_version\":
\"1.55.4\"}",
            "Version": 73,
            "LastModifiedDate": "2021-10-06T16:35:10.004000-07:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/ecs/optimized-ami/
amazon-linux-2/recommended",
            "DataType": "text"
        }
    ],
    "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 Amazon Machine Image (AMI) optimisée Amazon EKS en tant que paramètre public. Nous vous encourageons 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 Amazon EKS pour Amazon Linux 2.

Linux & macOS

```
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-08984d8491de17ca0\",
\"image_name\":\"amazon-eks-node-1.14-v20201007\",\"release_version\":
\"1.14.9-20201007\"}",
            "Version": 24,
            "LastModifiedDate": "2020-11-17T10:16:09.971000-08:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/eks/optimized-
ami/1.14/amazon-linux-2/recommended",
            "DataType": "text"
        }
    ],
    "InvalidParameters": []
}
```

Appel de paramètres publics pour les régions Services AWS, les points de terminaison, les zones de disponibilité, les zones locales et les zones de longueur d'onde

Vous pouvez appeler les zones de service Région AWS, de point de terminaison, de disponibilité et de longueur d'onde des paramètres publics en utilisant le chemin suivant.

`/aws/service/global-infrastructure`

> **Note**
>
> Actuellement, le chemin `/aws/service/global-infrastructure` n'est pris en charge Régions AWS que pour les requêtes suivantes :
>
> - USA Est (Virginie du Nord) (us-east-1)
> - USA Est (Ohio) (us-east-2)
> - US Ouest (N. California) (us-west-1)
> - USA Ouest (Oregon) (us-west-2)
> - Asie-Pacifique (Hong Kong) (ap-east-1)
> - Asie-Pacifique (Mumbai) (ap-south-1)
> - 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)
>
> Si vous travaillez dans une autre région commerciale, vous pouvez spécifier une région prise en charge dans votre requête pour afficher les résultats. Par exemple, si vous travaillez dans la région du Canada Ouest (Calgary) (ca-west-1), vous pouvez spécifier Canada (Centre) (ca-central-1) dans votre requête :
>
> ```
> aws ssm get-parameters-by-path \
>     --path /aws/service/global-infrastructure/regions \
>     --region ca-central-1
> ```

Afficher actif Régions AWS

Vous pouvez afficher la liste de tous les actifs à Régions AWS l'aide de la commande suivante dans le AWS Command Line Interface (AWS CLI).

Linux & macOS

```
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-south-2",
    "/aws/service/global-infrastructure/regions/ca-central-1",
    "/aws/service/global-infrastructure/regions/eu-central-2",
    "/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-southeast-1",
    "/aws/service/global-infrastructure/regions/ap-southeast-2",
    "/aws/service/global-infrastructure/regions/ap-southeast-3",
    "/aws/service/global-infrastructure/regions/cn-north-1",
    "/aws/service/global-infrastructure/regions/cn-northwest-1",
    "/aws/service/global-infrastructure/regions/eu-south-1",
    "/aws/service/global-infrastructure/regions/eu-south-2",
    "/aws/service/global-infrastructure/regions/us-east-2",
    "/aws/service/global-infrastructure/regions/us-west-1",
    "/aws/service/global-infrastructure/regions/ap-northeast-1",
```

```
    "/aws/service/global-infrastructure/regions/ap-south-1",
    "/aws/service/global-infrastructure/regions/ap-southeast-4",
    "/aws/service/global-infrastructure/regions/ca-west-1",
    "/aws/service/global-infrastructure/regions/eu-central-1",
    "/aws/service/global-infrastructure/regions/il-central-1",
    "/aws/service/global-infrastructure/regions/me-central-1",
    "/aws/service/global-infrastructure/regions/me-south-1",
    "/aws/service/global-infrastructure/regions/sa-east-1",
    "/aws/service/global-infrastructure/regions/us-gov-east-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 disponible Services AWS

Vous pouvez consulter la liste complète de toutes les options disponibles Services AWS et les trier par ordre alphabétique à l'aide de la commande suivante. Cet exemple de sortie a été tronqué faute d'espace.

Linux & macOS

```
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. Cet exemple a été tronqué faute d'espace.

```
[
     "/aws/service/global-infrastructure/services/accessanalyzer",
    "/aws/service/global-infrastructure/services/account",
    "/aws/service/global-infrastructure/services/acm",
    "/aws/service/global-infrastructure/services/acm-pca",
    "/aws/service/global-infrastructure/services/ahl",
```

```
    "/aws/service/global-infrastructure/services/aiq",
    "/aws/service/global-infrastructure/services/amazonlocationservice",
    "/aws/service/global-infrastructure/services/amplify",
    "/aws/service/global-infrastructure/services/amplifybackend",
    "/aws/service/global-infrastructure/services/apigateway",
    "/aws/service/global-infrastructure/services/apigatewaymanagementapi",
    "/aws/service/global-infrastructure/services/apigatewayv2",
    "/aws/service/global-infrastructure/services/appconfig",
    "/aws/service/global-infrastructure/services/appconfigdata",
    "/aws/service/global-infrastructure/services/appflow",
    "/aws/service/global-infrastructure/services/appintegrations",
    "/aws/service/global-infrastructure/services/application-autoscaling",
    "/aws/service/global-infrastructure/services/application-insights",
    "/aws/service/global-infrastructure/services/applicationcostprofiler",
    "/aws/service/global-infrastructure/services/appmesh",
    "/aws/service/global-infrastructure/services/apprunner",
    "/aws/service/global-infrastructure/services/appstream",
    "/aws/service/global-infrastructure/services/appsync",
    "/aws/service/global-infrastructure/services/aps",
    "/aws/service/global-infrastructure/services/arc-zonal-shift",
    "/aws/service/global-infrastructure/services/artifact",
    "/aws/service/global-infrastructure/services/athena",
    "/aws/service/global-infrastructure/services/auditmanager",
    "/aws/service/global-infrastructure/services/augmentedairuntime",
    "/aws/service/global-infrastructure/services/aurora",
    "/aws/service/global-infrastructure/services/autoscaling",
    "/aws/service/global-infrastructure/services/aws-appfabric",
    "/aws/service/global-infrastructure/services/awshealthdashboard",
```

Afficher les régions prises en charge pour un Service AWS

Vous pouvez consulter la liste des Régions AWS endroits où un service est disponible. Cet exemple utilise AWS Systems Manager (`ssm`).

Linux & macOS

```
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",
    "eu-central-1",
    "eu-central-2",
    "eu-west-1",
    "eu-west-2",
    "eu-west-3",
    "il-central-1",
    "me-south-1",
    "us-east-2",
    "us-gov-west-1",
    "af-south-1",
    "ap-northeast-3",
    "ap-southeast-1",
    "ap-southeast-4",
    "ca-central-1",
    "ca-west-1",
    "cn-north-1",
    "eu-north-1",
    "eu-south-2",
    "us-west-1",
    "ap-east-1",
    "ap-northeast-1",
    "ap-northeast-2",
    "ap-southeast-2",
    "ap-southeast-3",
    "cn-northwest-1",
    "eu-south-1",
    "me-central-1",
    "us-gov-east-1",
    "us-west-2",
    "ap-south-2",
    "sa-east-1",
    "us-east-1"
]
```

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. Cette commande interroge la région USA Est (Ohio) (us-east-2).

Linux & macOS

```
aws ssm get-parameter \
    --name /aws/service/global-infrastructure/regions/us-east-2/services/ssm/
endpoint \
    --query 'Parameter.Value'
```

Windows

```
aws ssm get-parameter ^
    --name /aws/service/global-infrastructure/regions/us-east-2/services/ssm/
endpoint ^
    --query Parameter.Value
```

La commande renvoie des informations telles que les suivantes.

```
"ssm.us-east-2.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 & macOS

```
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/afs1-az3",
            "Type": "String",
            "Value": "afs1-az3",
            "Version": 1,
            "LastModifiedDate": "2020-04-21T12:05:35.375000-04:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/
availability-zones/afs1-az3",
            "DataType": "text"
        },
        {
            "Name": "/aws/service/global-infrastructure/availability-zones/aps1-az2",
            "Type": "String",
            "Value": "aps1-az2",
            "Version": 1,
            "LastModifiedDate": "2020-04-03T16:13:57.351000-04:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/
availability-zones/aps1-az2",
            "DataType": "text"
        },
        {
            "Name": "/aws/service/global-infrastructure/availability-zones/apse3-az1",
            "Type": "String",
            "Value": "apse3-az1",
            "Version": 1,
            "LastModifiedDate": "2021-12-13T08:51:38.983000-05:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/
availability-zones/apse3-az1",
            "DataType": "text"
        }
    ]
}
```

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 & macOS

```
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-2`, dans cet exemple) à l'aide de la commande suivante.

Linux & macOS

```
aws ssm get-parameters-by-path \
    --path /aws/service/global-infrastructure/regions/us-east-2/availability-zones \
    --query 'Parameters[].Name | sort(@)'
```

Windows

```
aws ssm get-parameters-by-path ^
    --path /aws/service/global-infrastructure/regions/us-east-2/availability-zones ^
    --query "Parameters[].Name | sort(@)"
```

La commande renvoie des informations telles que les suivantes.

```
[
    "/aws/service/global-infrastructure/regions/us-east-2/availability-zones/use2-az1",
    "/aws/service/global-infrastructure/regions/us-east-2/availability-zones/use2-az2",
    "/aws/service/global-infrastructure/regions/us-east-2/availability-zones/use2-az3"
```

Afficher les ARN de la zone de disponibilité uniquement

Vous pouvez afficher les Amazon Resource Names (ARN) des zones de disponibilité uniquement à l'aide de la commande suivante.

Linux & macOS

```
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. Cet exemple a été tronqué faute d'espace.

```
[
    "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.

Linux & macOS

```
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. Cet exemple a été tronqué faute d'espace.

```
{
    "Parameters": [
        {
            "Name": "/aws/service/global-infrastructure/local-zones/afs1-los1-az1",
            "Type": "String",
            "Value": "afs1-los1-az1",
            "Version": 1,
            "LastModifiedDate": "2023-01-25T11:53:11.690000-05:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/
local-zones/afs1-los1-az1",
            "DataType": "text"
        },
        {
            "Name": "/aws/service/global-infrastructure/local-zones/apne1-tpe1-az1",
            "Type": "String",
            "Value": "apne1-tpe1-az1",
            "Version": 1,
            "LastModifiedDate": "2024-03-15T12:35:41.076000-04:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/
local-zones/apne1-tpe1-az1",
            "DataType": "text"
```

```
        },
        {
            "Name": "/aws/service/global-infrastructure/local-zones/aps1-ccu1-az1",
            "Type": "String",
            "Value": "aps1-ccu1-az1",
            "Version": 1,
            "LastModifiedDate": "2022-12-19T11:34:43.351000-05:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/
local-zones/aps1-ccu1-az1",
            "DataType": "text"
        }
    ]
}
```

Afficher les détails de la zone Wavelength

Vous pouvez afficher les zones Wavelength à l'aide de la commande suivante.

Linux & macOS

```
aws ssm get-parameters-by-path \
    --path /aws/service/global-infrastructure/wavelength-zones
```

Windows

```
aws ssm get-parameters-by-path ^
    --path /aws/service/global-infrastructure/wavelength-zones
```

La commande renvoie des informations telles que les suivantes. Cet exemple a été tronqué faute d'espace.

```
{
    "Parameters": [
        {
            "Name": "/aws/service/global-infrastructure/wavelength-zones/apne1-wl1-nrt-
wlz1",
            "Type": "String",
            "Value": "apne1-wl1-nrt-wlz1",
            "Version": 3,
            "LastModifiedDate": "2020-12-15T17:16:04.715000-05:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/
wavelength-zones/apne1-wl1-nrt-wlz1",
```

```
            "DataType": "text"
        },
        {
            "Name": "/aws/service/global-infrastructure/wavelength-zones/apne2-wl1-sel-
wlz1",
            "Type": "String",
            "Value": "apne2-wl1-sel-wlz1",
            "Version": 1,
            "LastModifiedDate": "2022-05-25T12:29:13.862000-04:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/
wavelength-zones/apne2-wl1-sel-wlz1",
            "DataType": "text"
        },
        {
            "Name": "/aws/service/global-infrastructure/wavelength-zones/cac1-wl1-yto-
wlz1",
            "Type": "String",
            "Value": "cac1-wl1-yto-wlz1",
            "Version": 1,
            "LastModifiedDate": "2022-04-26T09:57:44.495000-04:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/
wavelength-zones/cac1-wl1-yto-wlz1",
            "DataType": "text"
        }
    ]
}
```

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 & macOS

```
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/use1-bos1-az1"
```

La commande renvoie des informations telles que les suivantes. Cet exemple a été tronqué faute d'espace.

```
{
    "Parameters": [
        {
            "Name": "/aws/service/global-infrastructure/local-zones/use1-bos1-az1/
geolocationCountry",
            "Type": "String",
            "Value": "US",
            "Version": 3,
            "LastModifiedDate": "2020-12-15T14:16:17.641000-08:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/
local-zones/use1-bos1-az1/geolocationCountry",
            "DataType": "text"
        },
        {
            "Name": "/aws/service/global-infrastructure/local-zones/use1-bos1-az1/
geolocationRegion",
            "Type": "String",
            "Value": "US-MA",
            "Version": 3,
            "LastModifiedDate": "2020-12-15T14:16:17.794000-08:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/
local-zones/use1-bos1-az1/geolocationRegion",
            "DataType": "text"
        },
        {
            "Name": "/aws/service/global-infrastructure/local-zones/use1-bos1-az1/
location",
            "Type": "String",
            "Value": "US East (Boston)",
            "Version": 1,
            "LastModifiedDate": "2021-01-11T10:53:24.634000-08:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/
local-zones/use1-bos1-az1/location",
            "DataType": "text"
        },
        {
            "Name": "/aws/service/global-infrastructure/local-zones/use1-bos1-az1/
network-border-group",
            "Type": "String",
            "Value": "us-east-1-bos-1",
```

```
            "Version": 3,
            "LastModifiedDate": "2020-12-15T14:16:20.641000-08:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/
local-zones/use1-bos1-az1/network-border-group",
            "DataType": "text"
        },
        {
            "Name": "/aws/service/global-infrastructure/local-zones/use1-bos1-az1/
parent-availability-zone",
            "Type": "String",
            "Value": "use1-az4",
            "Version": 3,
            "LastModifiedDate": "2020-12-15T14:16:20.834000-08:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/
local-zones/use1-bos1-az1/parent-availability-zone",
            "DataType": "text"
        },
        {
            "Name": "/aws/service/global-infrastructure/local-zones/use1-bos1-az1/
parent-region",
            "Type": "String",
            "Value": "us-east-1",
            "Version": 3,
            "LastModifiedDate": "2020-12-15T14:16:20.721000-08:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/
local-zones/use1-bos1-az1/parent-region",
            "DataType": "text"
        },
        {
            "Name": "/aws/service/global-infrastructure/local-zones/use1-bos1-az1/zone-
group",
            "Type": "String",
            "Value": "us-east-1-bos-1",
            "Version": 3,
            "LastModifiedDate": "2020-12-15T14:16:17.983000-08:00",
            "ARN": "arn:aws:ssm:us-east-2::parameter/aws/service/global-infrastructure/
local-zones/use1-bos1-az1/zone-group",
            "DataType": "text"
        }
    ]
}
```

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 & macOS

```
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/use1-bos1-az1 ^
    --query "Parameters[].Name | sort(@)"
```

La commande renvoie des informations telles que les suivantes.

```
[
    "/aws/service/global-infrastructure/local-zones/use1-bos1-az1/geolocationCountry",
    "/aws/service/global-infrastructure/local-zones/use1-bos1-az1/geolocationRegion",
    "/aws/service/global-infrastructure/local-zones/use1-bos1-az1/location",
    "/aws/service/global-infrastructure/local-zones/use1-bos1-az1/network-border-
group",
    "/aws/service/global-infrastructure/local-zones/use1-bos1-az1/parent-availability-
zone",
    "/aws/service/global-infrastructure/local-zones/use1-bos1-az1/parent-region",
    "/aws/service/global-infrastructure/local-zones/use1-bos1-az1/zone-group"
]
```

Procédures Parameter Store

La démonstration décrite dans cette section vous montre comment créer, stocker et exécuter des paramètres avec Parameter Store, une des fonctionnalités de AWS Systems Manager, 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 de plus amples informations, veuillez consulter Qu'est-ce qu'un paramètre ?.

Table des matières

- Utiliser des paramètres Parameter Store dans Amazon Elastic Kubernetes Service

Création d'un paramètre SecureString et association d'un nœud à un domaine (PowerShell)

Cette démonstration illustre la manière d'associer un nœud Windows Server à un domaine à l'aide de paramètres AWS Systems Manager `SecureString` et Run Command. 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'une Clé gérée par AWS et transmise en tant que chaîne chiffrée.

Prérequis

Cette démonstration 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 en savoir plus, consultez Utilisation des jeux d'options DHCP dans le Guide de l'utilisateur Amazon VPC.

Pour créer un paramètre **`SecureString`** et associer un nœud à un domaine

1. Entrez les paramètres dans le système à l'aide des AWS Tools for Windows PowerShell.

   Dans les commandes suivantes, remplacez chaque *`espace réservé pour l'entrée utilisateur`* par vos propres informations.

   ```
   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 paramètres, les descriptions et d'autres propriétés ne sont pas chiffrés.

2. Attachez les politiques AWS Identity and Access Management (IAM) suivantes aux autorisations de rôle IAM pour votre nœud :

   - AmazonSSMManagedInstanceCore – Obligatoire. Cette politique gérée par AWS permet à une nœud géré d'utiliser les fonctions de base du service Systems Manager.

- AmazonSSMDirectoryServiceAccess – Obligatoire. Cette politique gérée par AWS permet à SSM Agent d'accéder à AWS Directory Service en votre nom pour les demandes d'association du domaine par le nœud géré.
- Une politique personnalisée pour l'accès au compartiment S3 – Obligatoire. SSM Agent, qui se trouve sur votre nœud et effectue des tâches Systems Manager, a besoin d'accéder à des compartiments Amazon Simple Storage Service (Amazon S3) spécifiques appartenant à Amazon. Dans la politique 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 pour des commandes Run Command ou des sessions Session Manager dans un compartiment S3, puis utiliser cette sortie ultérieurement aux 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'un référentiel de correctifs est appliquée.

  Pour obtenir des informations sur la création d'une politique personnalisée pour l'accès à un compartiment Amazon S3, veuillez consulter Créer une politique de compartiment S3 personnalisée pour un profil d'instance

  > **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 plus d'informations, consultez Création d'un compartiment dans le Guide de l'utilisateur Amazon Simple Storage Service.

- CloudWatchAgentServerPolicy – Facultatif. Cette politique gérée AWS vous permet d'exécuter l'agent CloudWatch sur des nœuds gérés. Cette politique permet de lire les informations sur un nœud et de les écrire dans Amazon CloudWatch. Votre profil d'instance n'a besoin de cette politique que si vous utilisez des services tels qu'Amazon EventBridge ou CloudWatch Logs.

  > **Note**
  >
  > L'utilisation des fonctions CloudWatch et EventBridge est facultative, mais nous vous recommandons de les configurer au début de votre processus de configuration

de Systems Manager si vous avez décidé de les utiliser. Pour plus d'informations, consultez le Guide de l'utilisateur Amazon EventBridge et le Guide de l'utilisateur Amazon CloudWatch Logs.

3. Modifiez le rôle IAM attaché au nœud et ajoutez la politique ci-dessous. Cette politique autorise le nœud à appeler l'API `kms:Decrypt` et `ssm:CreateDocument`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt",
                "ssm:CreateDocument"
            ],
            "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 en tant que `JoinInstanceToDomain.json` à l'emplacement suivant : `c:\temp\JoinInstanceToDomain.json`.

```
{
    "schemaVersion": "2.2",
    "description": "Run a PowerShell script to securely join a Windows Server
 instance to a domain",
    "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 \"Instance joined to domain successfully.
 Restarting\"; exit 3010}else{Write-Host \"Instance failed to join domain with
 error:\" $domainjoinerror; exit 1 }"
                ]
            }
        }
    ]
}
```

5. Exécutez la commande suivante dans Tools for 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 Tools for Windows PowerShell pour associer le nœud 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.
```

Utiliser des paramètres Parameter Store dans Amazon Elastic Kubernetes Service

Pour afficher les secrets de Secrets Manager et les paramètres Parameter Store sous forme de fichiers montés dans des pods Amazon EKS, vous pouvez utiliser le fournisseur de AWS secrets et de configuration (ASCP) pour le pilote CSI Kubernetes Secrets Store. (Parameter Storeest une capacité de AWS Systems Manager.) L'ASCP fonctionne avec Amazon Elastic Kubernetes Service (Amazon EKS) 1.17+. AWS Fargate (Fargate) les groupes de nœuds ne sont pas pris en charge.

Avec l'ASCP, vous pouvez récupérer des paramètres stockés et gérés dans Parameter Store. Vous pouvez ensuite utiliser les paramètres dans vos applications exécutées sur Amazon EKS. Si votre paramètre contient plusieurs paires de clé-valeur au format JSON, vous pouvez choisir celles à monter dans Amazon EKS. L'ASCP peut utiliser la syntaxe JMESPath pour interroger les paires clé-valeur dans votre paramètre.

Vous pouvez utiliser des rôles et des politiques AWS Identity and Access Management (IAM) pour limiter l'accès à vos paramètres à des pods Amazon EKS spécifiques d'un cluster. L'ASCP récupère l'identité du pod et l'échange contre un rôle IAM. L'ASCP assume le rôle IAM du pod. Ensuite, il peut récupérer à partir de Parameter Store les paramètres qui sont autorisés à assumer ce rôle.

Pour apprendre comment intégrer Secrets Manager à Amazon EKS, veuillez consulter Utilisation des secrets Secrets Manager dans Amazon Elastic Kubernetes Service.

Installation de l'ASCP

L'ASCP est disponible GitHub dans le référentiel secrets-store-csi-driver-provider-aws. Le référentiel contient également des exemples de fichiers YAML pour créer et monter un secret. Vous installez d'abord le pilote CSI Kubernetes Secrets Store, puis l'ASCP.

Pour installer le pilote CSI Kubernetes Secrets Store et l'ASCP

1. Pour installer le pilote CSI Kubernetes Secrets Store, exécutez les commandes suivantes. Pour obtenir les instructions d'installation complètes, consultez Installation dans le livre de pilotes CSI Secrets Store Kubernetes. Pour obtenir des informations sur l'installation de Helm, veuillez consulter Utilisation de Helm avec Amazon EKS.

```
helm repo add secrets-store-csi-driver https://kubernetes-sigs.github.io/secrets-store-csi-driver/charts
helm install -n kube-system csi-secrets-store secrets-store-csi-driver/secrets-store-csi-driver
```

2. Pour installer l'ASCP, utilisez le fichier YAML dans le répertoire de déploiement du GitHub référentiel. Pour plus d'informations sur l'installation de `kubectl`, consultez Installation de `kubectl`.

```
kubectl apply -f https://raw.githubusercontent.com/aws/secrets-store-csi-driver-provider-aws/main/deployment/aws-provider-installer.yaml
```

Étape 1 : configurer le contrôle d'accès

Pour accorder à votre pod Amazon EKS l'accès aux paramètres dans Parameter Store, vous devez d'abord créer une politique qui limite l'accès aux paramètres auxquels le pod doit accéder. Ensuite, vous créez un Rôle IAM pour le compte de service et vous lui attachez la politique. Pour de plus amples informations sur la restriction de l'accès aux paramètres Systems Manager à l'aide de politiques IAM, veuillez consulter Restriction de l'accès aux paramètres Systems Manager à l'aide des politiques IAM.

> **Note**
>
> Lorsque vous utilisez les paramètres Parameter Store, l'autorisation `ssm:GetParameters` doit être incluse dans la politique.

L'ASCP récupère l'identité du pod et l'échange pour un rôle IAM. L'ASCP assume le rôle IAM du pod, ce qui lui donne accès aux paramètres que vous avez autorisés. Les autres conteneurs ne peuvent pas accéder aux paramètres sauf si vous les associez également au rôle IAM.

Étape 2 : monter les paramètres dans Amazon EKS

Pour afficher des paramètres dans Amazon EKS comme s'il s'agissait de fichiers sur le système de fichiers, créez un fichier YAML `SecretProviderClass` qui contient des informations sur vos paramètres et sur comment les monter dans le pod Amazon EKS.

La `SecretProviderClass` doit se trouver dans le même espace de noms que le pod Amazon EKS auquel il fait référence.

`SecretProviderClass`

Le fichier YAML `SecretProviderClass` a le format suivant.

```
apiVersion: secrets-store.csi.x-k8s.io/v1alpha1
kind: SecretProviderClass
metadata:
   name: <NAME>
spec:
  provider: aws
  parameters:
```

parameters

Contient les détails de la demande de montage.

objects

Chaîne contenant une déclaration YAML des paramètres à monter. Nous vous recommandons d'utiliser une chaîne YAML multi-ligne ou un caractère pipe (|).

objectName

Nom convivial du paramètre. Il devient le nom de fichier du paramètre dans le pod Amazon EKS sauf si vous spécifiez `objectAlias`. Pour Parameter Store, il doit s'agir du `Name` du paramètre, et non d'un Amazon Resource Name (ARN) complet.

jmesPath

(Facultatif) Une carte des clés du paramètre codé JSON avec les fichiers à monter dans Amazon EKS. L'exemple suivant montre comment se présente un paramètre codé JSON.

```
{
    "username" : "myusername",
    "password" : "mypassword"
}
```

Les clés sont `username` et `password`. La valeur associée à `username` est `myusername` et la valeur associée à `password` est `mypassword`.

path

La clé dans le paramètre.

objectAlias

Le nom du fichier à monter dans le pod Amazon EKS.

objectType

Pour Parameter Store, ce champ est obligatoire. Utilisez `ssmparameter`.

objectAlias

(Facultatif) Nom de fichier du paramètre dans le pod Amazon EKS. Si vous ne spécifiez pas ce champ, `objectName` apparaît en tant que nom de fichier.

objectVersion

(Facultatif) Numéro de version du paramètre. Nous vous recommandons de ne pas utiliser ce champ car vous devez le mettre à jour chaque fois que vous mettez à jour le paramètre. Par défaut, la version la plus récente est utilisée. Pour les paramètres Parameter Store, vous pouvez utiliser `objectVersion` ou `objectVersionLabel`, mais pas les deux.

objectVersionLabel

(Facultatif) Étiquette du paramètre pour la version. La version par défaut est la plus récente. Pour les paramètres Parameter Store, vous pouvez utiliser `objectVersion` ou `objectVersionLabel`, mais pas les deux.

region

(Facultatif) Le Région AWS du paramètre. Si vous n'utilisez pas ce champ, ASCP va rechercher la région à partir de l'annotation sur le nœud. Comme cette recherche ajoute une surcharge aux demandes de montage, nous vous recommandons de fournir la région pour les clusters qui utilisent un grand nombre de pods.

pathTranslation

(Facultatif) Un caractère de substitution unique à utiliser si le nom de fichier (`objectName` ou `objectAlias`) contient le caractère séparateur de chemin, tel que la barre oblique (/) sous Linux. Si un nom de paramètre contient le séparateur de chemin, l'ASCP ne peut pas créer un fichier monté avec ce nom. Au lieu de cela, vous pouvez remplacer le caractère séparateur de chemin d'accès par un autre caractère en le saisissant dans ce champ. Si vous n'utilisez pas ce champ, la valeur par défaut est le trait de soulignement (_). Par exemple,`My/Path/Parameter` se monte en tant que `My_Path_Parameter`.

Pour empêcher la substitution de caractères, entrez la chaîne `False`.

Exemple

L'exemple de configuration suivant illustre une `SecretProviderClass` avec une ressource de paramètre Parameter Store.

```
apiVersion: secrets-store.csi.x-k8s.io/v1alpha1
kind: SecretProviderClass
metadata:
    name: aws-secrets
spec:
    provider: aws
    parameters:
        objects: |
            - objectName: "MyParameter"
              objectType: "ssmparameter"
```

Étape 3 : mettre à jour le déploiement YAML

Mettez à jour votre déploiement YAML pour utiliser le pilote `secrets-store.csi.k8s.io` et référencer la ressource `SecretProviderClass` créée à l'étape précédente. Cela garantit que votre cluster utilise le pilote CSI Secrets Store.

Voici un exemple de déploiement YAML avec une `SecretProviderClass` nommée `aws-secrets`.

```
volumes:
  - name: secrets-store-inline
    csi:
      driver: secrets-store.csi.k8s.io
      readOnly: true
      volumeAttributes:
        secretProviderClass: "aws-secrets"
```

Didacticiel : Créer et monter un paramètre dans un pod Amazon EKS

Dans ce didacticiel, vous créez un exemple de paramètre dansParameter Store, puis vous montez le paramètre dans un pod Amazon EKS et vous le déployez.

Avant de commencer, installez l'ASCP. Pour plus d'informations, consultez the section called "Installation de l'ASCP".

Pour créer et monter un secret

1. Définissez le nom Région AWS et le nom de votre cluster en tant que variables shell afin de pouvoir les utiliser dans les `bash` commandes. Pour *`la région`*, entrez l' Région AWS endroit où s'exécute votre cluster Amazon EKS. Pour *`clustername`*, saisissez un nom pour votre cluster.

```
REGION=region
CLUSTERNAME=clustername
```

2. Créez un paramètre de test.

```
aws ssm put-parameter --name "MyParameter" --value "EKS parameter" --type String --
region "$REGION"
```

3. Créez une politique de ressource pour le pod qui limite son accès au paramètre que vous avez créé à l'étape précédente. Pour *`parameter-arn`*, utilisez l'ARN du paramètre. Enregistrez l'ARN de politique dans une variable shell. Pour récupérer l'ARN du paramètre, utilisez `get-parameter`.

```
POLICY_ARN=$(aws --region "$REGION" --query Policy.Arn --output text iam create-
policy --policy-name nginx-parameter-deployment-policy --policy-document '{
    "Version": "2012-10-17",
    "Statement": [ {
        "Effect": "Allow",
        "Action": ["ssm:GetParameter", "ssm:GetParameters"],
        "Resource": ["parameter-arn"]
    } ]
}')
```

4. Créez un fournisseur OpenID Connect (OIDC) IAM pour le cluster si vous n'en avez pas déjà un. Pour de plus amples informations, veuillez consulter Créer un fournisseur IAM OIDC pour votre cluster.

```
eksctl utils associate-iam-oidc-provider --region="$REGION" --
cluster="$CLUSTERNAME" --approve # Only run this once
```

5. Créez le compte de service utilisé par le pod et associez la politique de ressources que vous avez créée à l'étape 3 à ce compte de service. Pour ce didacticiel, pour le nom du compte de service, vous utilisez nginx-deployment-sa. Pour de plus amples informations, consultez Création d'un rôle IAM pour votre compte de service.

```
eksctl create iamserviceaccount --name nginx-deployment-sa --region="$REGION" --
cluster "$CLUSTERNAME" --attach-policy-arn "$POLICY_ARN" --approve --override-
existing-serviceaccounts
```

6. Créez la `SecretProviderClass` pour spécifier le paramètre à monter dans le pod. La commande suivante utilise l'emplacement de fichier d'un fichier `SecretProviderClass` nommé `ExampleSecretProviderClass.yaml`. Pour plus d'informations sur la création de votre propre `SecretProviderClass`, consultez the section called "`SecretProviderClass`".

```
kubectl apply -f ./ExampleSecretProviderClass.yaml
```

7. Déployez votre pod La commande suivante utilise un fichier de déploiement nommé `ExampleDeployment.yaml`. Pour plus d'informations sur la création de votre propre `SecretProviderClass`, consultez the section called "Étape 3 : mettre à jour le déploiement YAML".

```
kubectl apply -f ./ExampleDeployment.yaml
```

8. Pour vérifier que le paramètre a été monté correctement, utilisez la commande suivante et confirmez que votre valeur de paramètre apparaît.

```
kubectl exec -it $(kubectl get pods | awk '/nginx-deployment/{print $1}' | head -1)
 cat /mnt/secrets-store/MyParameter; echo
```

   La valeur du paramètre apparaît.

```
"EKS parameter"
```

Résolution des problèmes

Vous pouvez afficher la plupart des erreurs en décrivant le déploiement du pod.

Pour afficher les messages d'erreur pour votre conteneur

1. Obtenez une liste de noms de pod à l'aide de la commande suivante. Si vous n'utilisez pas l'espace de noms par défaut, utilisez `-n <NAMESPACE>`.

   ```
   kubectl get pods
   ```

2. Pour décrire le pod, dans la commande suivante, pour *pod-id*, utilisez l'ID de pod des pods trouvés à l'étape précédente. Si vous n'utilisez pas l'espace de noms par défaut, utilisez `-n <NAMESPACE>`.

   ```
   kubectl describe pod/pod-id
   ```

Pour voir les erreurs pour l'ASCP

- Pour trouver plus d'informations dans les journaux du fournisseur, dans la commande suivante, pour *pod-id*, utilisez l'ID du pod csi-secrets-store-provider-aws.

  ```
  kubectl -n kube-system get pods
  kubectl -n kube-system logs pod/pod-id
  ```

Audit et journalisation de l'activité de Parameter Store

AWS CloudTrail capture les appels d'API effectués dans la console AWS Systems Manager, l'AWS Command Line Interface (AWS CLI) et le kit SDK Systems Manager. Vous pouvez afficher les informations dans la console CloudTrail ou dans un compartiment Amazon Simple Storage Service (Amazon S3). Tous les journaux CloudTrail de votre compte utilisent un compartiment. 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' AWS Systems Manager API avec AWS CloudTrail. Pour de plus amples informations sur les options d'audit et de journalisation de Systems Manager, veuillez consulter Surveillance AWS Systems Manager.

Résolution des problèmes de Parameter Store

Utilisez les informations suivantes pour vous aider à résoudre les problèmes liés Parameter Store à une fonctionnalité de AWS Systems Manager.

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`.

Aucune autorisation pour créer une instance

Problème : vous essayez de créer une instance à l'aide d'un `aws:ec2:image` paramètre, mais vous recevez un message d'erreur tel que « Vous n'êtes pas autorisé à effectuer cette opération ».

- Solution : Vous ne disposez pas de toutes les autorisations nécessaires pour créer une instance EC2 à l'aide d'une valeur de paramètre, telle que les autorisations pour `ec2:RunInstances`ec2:DescribeImages, et`ssm:GetParameter`, entre autres. Contactez un utilisateur disposant d'autorisations d'administrateur au sein de votre organisation pour demander les autorisations nécessaires.

EventBridge signale le message d'erreur « Impossible de décrire la ressource »

Problème : vous avez exécuté une commande pour créer un paramètre `aws:ec2:image`, mais la création de ce dernier a échoué. Vous recevez une notification d'Amazon EventBridge signalant l'exception « Impossible de décrire la ressource ».

Solution : ce message peut indiquer ce qui suit :

- Vous n'avez pas reçu l'autorisation requise pour l'opération d'API `ec2:DescribeImages`, ou vous n'avez pas l'autorisation d'accéder à l'image spécifique référencée dans le paramètre. Contactez un utilisateur disposant des autorisations d'administrateur dans votre organisation pour demander les autorisations nécessaires.
- L'ID d'Amazon Machine Image (AMI) que vous avez saisi en tant que valeur de paramètre n'est pas valide. Assurez-vous de saisir l'identifiant d'un AMI numéro disponible sur le compte courant Région AWS et sur lequel vous travaillez.

Le nouveau paramètre **aws:ec2:image** n'est pas disponible

Problème : vous venez d'exécuter une commande pour créer un paramètre `aws:ec2:image` 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 de données `aws:ec2:image`, un numéro de version est immédiatement généré pour le

paramètre, mais le format de ce dernier doit être validé avant qu'il ne 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 :

- EventBridge Utilisez-le pour vous envoyer des notifications concernant vos opérations `create` et les `update` paramètres. Ces notifications indiquent si une opération de paramètre a réussi ou non. Pour plus d'informations sur l'abonnement à Parameter Store des événements dans EventBridge, voirConfiguration de notifications ou d'actions de déclenchement basées sur des événements Parameter Store.
- Dans la section Parameter Store de la console Systems Manager, mettez à jour périodiquement la liste des paramètres pour rechercher 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 Command Line Interface (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, vous pouvez réessayer 5 minutes après la première tentative.

AWS Systems Manager Gestion du changement

AWS Systems Manager fournit les fonctionnalités suivantes pour apporter des modifications à vos AWS ressources.

Rubriques

AWS Systems Manager Change Manager

Change Manager, une fonctionnalité de AWS Systems Manager, est un cadre de gestion des modifications d'entreprise permettant de demander, d'approuver, de mettre en œuvre et de signaler les modifications opérationnelles apportées à la configuration et à l'infrastructure de vos applications. À partir d'un seul compte d'administrateur délégué, si vous en avez un AWS Organizations, vous pouvez gérer les modifications sur plusieurs comptes Comptes AWS et de manière transversale Régions AWS. En variante, en utilisant un compte local, vous pouvez gérer les modifications d'un Compte AWS unique. Change ManagerÀ utiliser pour gérer les modifications apportées aux AWS ressources et aux ressources locales. Pour vos premiers pas dans Change Manager, ouvrez Systems Manager console. Dans le panneau de navigation, sélectionnez Change Manager.

Avec Change Manager, vous pouvez utiliser des modèles de modifications approuvés pour automatiser les processus de modification de vos ressources et éviter les résultats non intentionnels lors des modifications opérationnelles. Chaque modèle de modification spécifie ce qui suit :

- Un ou plusieurs runbooks Automation à choisir par un utilisateur lors de la création d'une demande de modification. Les modifications apportées à vos ressources sont définies dans les runbooks Automation. Vous pouvez inclure des runbooks personnalisés ou des runbooks gérés par AWS dans les modèles de modifications que vous créez. Lorsqu'un utilisateur crée une demande de modification, il peut choisir quel runbook inclure dans la demande, parmi ceux qui sont disponibles. En outre, vous pouvez créer des modèles de modifications où l'utilisateur qui effectue la demande peut spécifier n'importe quel runbook dans la demande de modification.
- Les utilisateurs du compte qui doivent vérifier les demandes de modifications qui ont été faites à l'aide de ce modèle de modification.

- La rubrique Amazon Simple Notification Service (Amazon SNS) qui sert à informer les approbateurs affectés qu'une demande de modification est prête à être vérifiée.
- L' CloudWatch alarme Amazon utilisée pour surveiller le flux de travail du runbook.
- La rubrique Amazon SNS qui sert à envoyer des notifications relatives aux changements de statut des demandes de modifications créées à l'aide du modèle de modification.
- Les balises à appliquer au modèle de modification pour catégoriser et filtrer vos modèles de modifications.
- La possibilité que les demandes de modifications créées à partir du modèle de modification soient exécutées sans une étape d'approbation (demandes approuvées automatiquement).

Grâce à son intégrationChange Calendar, qui est une autre fonctionnalité de Systems Manager, elle vous permet Change Manager également de mettre en œuvre des modifications en toute sécurité tout en évitant les conflits de calendrier avec des événements commerciaux importants. Change Managerintégration avec AWS Organizations et AWS IAM Identity Center aide à gérer les changements au sein de votre organisation à partir d'un seul compte en utilisant votre système de gestion des identités existant. Vous pouvez surveiller la progression des modifications à partir de Change Manager et auditer les modifications opérationnelles au sein de votre organisation afin d'améliorer la visibilité et la responsabilisation.

Change Manager complète les contrôles de sécurité de vos pratiques d'intégration continue (IC) et votre méthodologie de livraison continue(CD). Change Manager ne concerne pas les modifications apportées dans le cadre d'un processus de publication automatisé, un pipeline CI/CD par exemple, sauf si une exception ou une approbation est exigée.

Fonctionnement d'Change Manager

Lorsque la nécessité d'une modification opérationnelle standard ou d'urgence est identifiée, une personne de l'organisation crée une demande de modification basée sur l'un des modèles de modifications créés pour votre organisation ou votre compte.

Si la modification demandée exige des approbations manuelles, Change Manager avertit les approbateurs désignés via une notification Amazon SNS qu'une demande de modification est prête à être vérifiée. Vous pouvez désigner des approbateurs pour les demandes de modifications dans le modèle de modification ou laisser les utilisateurs désigner eux-mêmes des approbateurs dans la demande de modification. Vous pouvez affecter différents vérificateurs à différents modèles. Par exemple, affectez un utilisateur, groupe d'utilisateurs ou rôle AWS Identity and Access Management

(IAM) à l'approbation des demandes de modification des nœuds gérés, et un autre utilisateur, groupe ou rôle IAM à celle des demandes de modification de la base de données. Si le modèle de modification autorise les approbations automatiques et qu'une politique d'utilisateur d'un demandeur ne l'interdit pas, l'utilisateur peut également choisir d'exécuter le runbook Automation pour sa demande sans une étape de vérification (à l'exception des événements de gel des modifications).

Pour chaque modèle de modification, vous pouvez ajouter jusqu'à cinq niveaux d'approbateurs. Par exemple, vous pouvez d'abord demander à des vérificateurs techniques d'approuver une demande de modification créée à partir d'un modèle de modification, puis exiger un second niveau d'approbation de la part d'un ou plusieurs responsables.

Change Manager est intégré à AWS Systems Manager Change Calendar. Lorsqu'une modification demandée est approuvée, le système détermine d'abord si la demande est en conflit avec d'autres activités métier planifiées. Si un conflit est détecté, Change Manager peut bloquer la modification ou exiger des approbations supplémentaires avant de démarrer le flux de travail du runbook. Par exemple, vous pouvez autoriser l'exécution de modifications uniquement pendant les heures ouvrables afin que les équipes soient disponibles pour gérer les problèmes inattendus. Pour l'exécution de modifications en dehors de ces heures, vous pouvez exiger l'approbation d'un échelon supérieur de la direction, servant alors d'approbateurs de gel des modifications. Pour des modifications d'urgence, Change Manager peut ignorer l'étape de vérification de conflits ou d'événements de blocage en lien avec Change Calendar, après qu'une demande de modification a été approuvée.

Lorsqu'il est temps d'implémenter un changement approuvé, Change Manager exécute le runbook Automation spécifié dans la demande de modification associée. Seules les opérations définies dans les demandes de modifications approuvées sont autorisées lors de l'exécution des flux de travail du runbook. Cette approche permet d'éviter les résultats non intentionnels durant l'implémentation des modifications.

En plus de limiter les modifications qui peuvent être apportées lors de l'exécution d'un flux de travail de runbook, Change Manager vous aide également à contrôler la simultanéité et les seuils d'erreur. Vous sélectionnez le nombre de ressources sur lesquelles un flux de travail de runbook peut s'exécuter simultanément, le nombre de comptes sur lesquels la modification peut s'exécuter simultanément, et le nombre d'échecs à autoriser avant que le processus soit arrêté, et (si le runbook inclut un script d'annulation), annulé. Vous pouvez également suivre la progression des modifications apportées à l'aide d' CloudWatch alarmes.

Une fois le flux de travail de runbook terminé, vous pouvez vérifier les détails des modifications apportées. Ces détails comprennent la raison de la demande de modification, le modèle de

modification utilisé, la personne qui a demandé et approuvé les modifications, et la façon dont les modifications ont été implémentées.

Plus d'informations

Présentation de AWS Systems ManagerChange Manager dans le Blog d'actualités AWS

Quels avantages Change Manager présente-t-il pour mes opérations ?

Les avantages d'Change Manager sont les suivants :

- Réduction des risques d'interruption de service et de temps d'arrêt

  Change Manager peut renforcer la sécurité des modifications opérationnelles en veillant à ce que seules les modifications approuvées soient implémentées lors de l'exécution d'un flux de travail de runbook. Vous pouvez bloquer les modifications non planifiées et non vérifiées. Change Manager vous aide à éviter les types de résultats non intentionnels dus à une erreur humaine et qui exigent des heures coûteuses de recherche et de retour sur trace.
- Obtention d'audits et de rapports détaillés sur les historiques de modifications

  Change Manager favorise la responsabilisation grâce à des rapports et des audits cohérents sur les modifications apportées au sein de votre organisation, l'intention des modifications et des détails sur les personnes qui les ont approuvées et implémentées.
- Évitement des conflits ou violations de planification

  Change Manager peut détecter des conflits de planification, tels que des événements de vacances ou des lancements de nouveaux produits, en se basant sur le calendrier des modifications actives de votre organisation. Vous pouvez autoriser l'exécution de workflows de runbook uniquement pendant les heures ouvrables ou uniquement avec des approbations supplémentaires.
- Adaptation des exigences de gestion des modifications au calendrier de votre entreprise

  En fonction des périodes d'activité, vous pouvez implémenter différentes exigences de gestion des modifications. Par exemple, au cours de la période des end-of-month rapports, de la saison des impôts ou d'autres périodes commerciales critiques, vous pouvez bloquer les modifications ou demander l'approbation du directeur pour les modifications susceptibles d'entraîner des risques opérationnels inutiles.
- Gestion centralisée des modifications entre comptes

Grâce à son intégration à Organizations, Change Manager vous permet de gérer les modifications dans toutes vos unités organisationnelles (UO) depuis un compte d'administrateur délégué unique. Vous pouvez activer Change Manager pour l'utiliser dans toute votre organisation ou seulement certaines unités organisationnelles.

À qui est destiné Change Manager ?

Change Managerconvient aux AWS clients et organisations suivants :

- Tout AWS client qui souhaite améliorer la sécurité et la gouvernance des modifications opérationnelles apportées à son environnement cloud ou sur site.
- Les organisations désireuses d'améliorer la collaboration et la visibilité entre équipes, d'accroître la disponibilité des applications en évitant les temps d'arrêt, et de réduire les risques associés aux tâches manuelles et répétitives.
- Les organisations qui doivent se conformer aux bonnes pratiques en matière de gestion des modifications.
- Les clients qui ont besoin d'un historique auditable des modifications apportées à la configuration et à l'infrastructure de leur application.

Quelles sont les principales fonctionnalités Change Manager ?

Les principales fonctionnalités de Change Manager sont décrites ci-après :

- Prise en charge intégrée des bonnes pratiques en matière de gestion des modifications

  Avec Change Manager, vous pouvez appliquer certaines bonnes pratiques de gestion des modifications à vos opérations. Vous pouvez choisir d'activer les options suivantes :

  - Vérifiez Change Calendar pour voir si des restrictions affectent des événements et que des modifications ne peuvent être effectuées que pendant les heures ouvrables.
  - Autorisez des modifications durant des événements restreints avec des approbations supplémentaires émanant d'approbateurs de gel des modifications.
  - Exiger que des CloudWatch alarmes soient spécifiées pour tous les modèles de modification.
  - Exigez que tous les modèles de modifications créés dans votre compte soient vérifiés et approuvés avant d'être utilisés pour créer des demandes de modifications.

- Différents chemins d'approbation pour les périodes civiles fermées ainsi que les demandes de modification d'urgence

  Vous pouvez autoriser une option qui vérifie les événements restreints dans Change Calendar et bloque les demandes de modifications approuvées jusqu'à ce que l'événement soit terminé. Vous pouvez également désigner un second groupe d'approbateurs, les approbateurs de gel des modifications, qui peuvent autoriser la modification même en dehors des heures ouvrables. Vous pouvez aussi créer des modèles de modifications d'urgence. Les demandes de modifications créées à partir d'un modèle de modification d'urgence exigent toujours des approbations régulières, mais ne sont pas soumises à des restrictions de calendrier et ne nécessitent pas d'approbations de gel des modifications.
- Contrôler la méthode et l'instant de démarrage de flux de travail de runbook

  Les flux de travail de runbook peuvent être démarrés de façon planifiée ou dès que les approbations sont terminées (sous réserve des règles de restriction de calendrier).
- Prise en charge intégrée des notifications

  Indiquez les personnes qui, au sein de votre organisation, doivent vérifier et approuver les modèles de modifications et les demandes de modifications. Affectez une rubrique Amazon SNS à un modèle de modification pour envoyer des notifications aux abonnés de la rubrique à propos des changements de statut de demandes de modifications créées avec ce modèle de modification.
- Intégration avec AWS Systems Manager Change Calendar

  Change Manager permet aux administrateurs de restreindre les modifications de planification pendant des périodes spécifiées. Par exemple, vous pouvez créer une politique qui autorise les modifications uniquement pendant les heures ouvrables afin que l'équipe soit disponible pour gérer les problèmes. Vous pouvez également restreindre les modifications lors d'événements professionnels importants. Par exemple, les entreprises de vente au détail peuvent restreindre les modifications lors d'événements de grande envergure. Vous pouvez également exiger des approbations supplémentaires pendant les périodes restreintes.
- Intégration avec Active Directory AWS IAM Identity Center et prise en charge

  Grâce à l'intégration à IAM Identity Center, les membres de votre organisation peuvent accéder à Comptes AWS et gérer leurs ressources en utilisant Systems Manager à partir d'une identité utilisateur commune. L'utilisation d'IAM Identity Center vous permet d'affecter à vos utilisateurs l'accès à des comptes dans AWS.

L'intégration à Active Directory permet d'affecter des utilisateurs de votre compte Active Directory en tant qu'approbateurs pour les modèles de modifications créés pour vos opérations Change Manager.

- Intégration aux CloudWatch alarmes Amazon

  Change Managerest intégré aux CloudWatch alarmes. Change Managerécoute les CloudWatch alarmes pendant le flux de travail du runbook et prend toutes les mesures, y compris l'envoi de notifications, définies pour l'alarme.

- Intégration avec AWS CloudTrail Lake

  En créant un magasin de données d'événements dans AWS CloudTrail Lake, vous pouvez consulter des informations vérifiables sur les modifications apportées par les demandes de modification exécutées dans votre compte ou votre organisation. Les informations stockées sur les événements incluent des informations telles que les suivantes :

  - Les actions d'API qui ont été exécutées
  - Les paramètres de demande inclus pour ces actions
  - L'utilisateur qui a exécuté l'action
  - Les ressources qui ont été mises à jour au cours du processus

- Intégration avec AWS Organizations

  L'utilisation des fonctionnalités intercompte proposées par Organizations vous permet d'utiliser un compte d'administrateur délégué pour gérer des opérations Change Manager dans des UO de votre organisation. Dans votre compte de gestion Organizations, vous pouvez spécifier quel compte doit être le compte d'administrateur délégué. Vous pouvez également contrôler dans laquelle de vos UO Change Manager peut être utilisé.

L'utilisation d'Change Manager entraîne-t-elle des frais ?

Oui Change Managerle prix est calculé sur une pay-per-use base. Vous ne payez que ce que vous utilisez. Pour plus d'informations, consultez AWS Systems Manager Pricing (Tarification CTlong).

Quels sont les principaux composants de Change Manager?

Voici les composants de Change Manager, que vous utilisez pour gérer le processus de modification dans votre organisation ou votre compte :

Compte administrateur délégué

Si vous utilisez Change Manager dans une organisation, vous utilisez un compte d'administrateur délégué. C'est le Compte AWS désigné comme compte pour la gestion des activités opérationnelles dans Systems Manager, notamment Change Manager. Le compte d'administrateur délégué gère les activités de modification au sein de votre organisation. Lorsque vous configurez votre organisation pour une utilisation avec Change Manager, vous spécifiez lequel de vos comptes tiendra ce rôle. Le compte d'administrateur délégué doit être le seul membre de l'unité d'organisation (UO) à laquelle il est affecté. Le compte d'administrateur délégué n'est pas obligatoire si vous ne l'utilisez qu'Change Manageravec un Compte AWS seul.

> **Important**
>
> Si vous utilisez Change Manager au sein d'une organisation, nous vous recommandons de toujours apporter les modifications à partir du compte d'administrateur délégué. Bien qu'il soit possible d'apporter des modifications à partir d'autres comptes de l'organisation, celles-ci ne seront pas signalées ou affichées à partir du compte d'administrateur délégué.

Modèle de modification

Un modèle de modification est un ensemble de paramètres de configuration dans Change Manager, qui définissent des éléments tels que les approbations requises, les runbooks disponibles et les options de notification relatives aux demandes de modifications.

Vous pouvez exiger que les modèles de modifications créés par les utilisateurs de votre organisation ou de votre compte passent par un processus d'approbation avant d'être utilisés.

Change Manager prend en charge deux types de modèles de modifications. Pour une demande de modification approuvée basée sur un modèle de modification d'urgence, la modification demandée peut être effectuée même s'il existe des événements de blocage dans Change Calendar. Pour une demande de modification approuvée basée sur un modèle de modification standard, la modification demandée ne peut pas être effectuée s'il existe des événements de blocage dans Change Calendar, sauf si des approbations supplémentaires sont reçues d'approbateurs d'événements de gel des modifications.

Demande de modification

Une demande de modification est une demande visant Change Manager à exécuter un runbook d'automatisation qui met à jour une ou plusieurs ressources dans votre environnement AWS ou dans votre environnement sur site. Une demande de modification est créée à l'aide d'un modèle de modification.

Lorsque vous créez une demande de modification, un ou plusieurs approbateurs de votre organisation ou de votre compte doivent la vérifier et l'approuver. Sans les approbations requises, le flux de travail de runbook, qui applique les modifications demandées, n'est pas autorisé à s'exécuter.

Dans le système, les demandes de modification sont un type OpsItem d'entrée AWS Systems Manager OpsCenter. Toutefois, des OpsItems du type `/aws/changerequest` ne sont pas affichés dans OpsCenter. En tant qu'OpsItems, les demandes de modification sont soumises aux mêmes quotas que pour les autres types d'OpsItems.

En outre, pour créer une demande de modification par programmation, vous n'appelez pas l'opération d'API `CreateOpsItem`. Vous utilisez plutôt l'opération d'API `StartChangeRequestExecution`. Cependant, la demande de modification ne peut pas s'exécuter immédiatement ; elle doit être approuvée, et il ne doit y avoir aucun événement de blocage dans Change Calendarsusceptible d'empêcher l'exécution du flux de travail. Après que les approbations ont été reçues, et si le calendrier n'est pas bloqué (ou que l'autorisation de contourner les événements de calendrier bloqués a été accordée), l'action `StartChangeRequestExecution` peut se terminer.

Flux de travail de runbook

Un flux de travail de runbook est le processus des modifications demandées apportées aux ressources ciblées dans votre environnement cloud ou local. Chaque demande de modification désigne un runbook Automation unique, à utiliser pour effectuer la modification demandée. Le flux de travail de runbook se produit après que toutes les approbations requises ont été accordées et s'il n'y a pas d'événements de blocage dans Change Calendar. Si la modification a été planifiée à une date et une heure spécifiques, le flux de travail de runbook ne démarre pas avant la date prévue, même si toutes les approbations ont été reçues et que le calendrier n'est pas bloqué.

Rubriques

- Configuration de Change Manager
- Utilisation des Change Manager
- Audit et journalisation de l'activité de Change Manager

- Résolution des problèmes de Change Manager

Configuration de Change Manager

Vous pouvez utiliser Change Manager, une fonctionnalité de AWS Systems Manager, pour gérer les modifications au sein d'une organisation, comme configuré dans AWS Organizations, ou pour un seul Compte AWS.

Si vous utilisez Change Manager avec une organisation, commencez par la rubrique Configuration de Change Manager pour une organisation (compte de gestion), puis passez à Configuration d'options et de bonnes pratiques Change Manager.

Si vous utilisez Change Manager avec un seul compte, passez directement à Configuration d'options et de bonnes pratiques Change Manager.

> **Note**
>
> Si vous commencez à utiliser Change Manager avec un seul compte, mais que ce compte est ajouté par la suite à une unité d'organisation pour laquelle Change Manager est autorisé, les paramètres de votre compte unique ne sont pas pris en compte.

Rubriques

- Configuration de Change Manager pour une organisation (compte de gestion)
- Configuration d'options et de bonnes pratiques Change Manager
- Configuration des rôles et des autorisations pour Change Manager
- Contrôler l'accès aux flux de travail de runbook d'approbation automatique

Configuration de Change Manager pour une organisation (compte de gestion)

Les tâches décrites dans cette rubrique s'appliquent si vous utilisez Change Manager une fonctionnalité de AWS Systems Manager, avec une organisation configurée dans AWS Organizations. Si vous ne souhaitez l'utiliser Change Manager qu'avec un seul Compte AWS, passez à la rubriqueConfiguration d'options et de bonnes pratiques Change Manager.

Effectuez les tâches décrites dans cette section dans un compte Compte AWS qui sert de compte de gestion dans Organizations. Pour obtenir des informations sur le compte de gestion et d'autres concepts Organizations, veuillez consulter Terminologie et concepts relatifs àAWS Organizations.

Si vous devez activer Organizations et spécifier votre compte en tant que compte de gestion avant de continuer, veuillez consulter Création et gestion d'une organisation dans le Guide de l'utilisateur AWS Organizations .

> **Note**
>
> Ce processus de configuration ne peut pas être effectué dans les cas suivants Régions AWS :
>
> - Europe (Milan) (eu-south-1)
> - Moyen-Orient (Bahreïn) (me-south-1)
> - Afrique (Le Cap) (af-south-1)
> - Asie-Pacifique (Hong Kong) (ap-east-1)
>
> Pour cette procédure, vérifiez que vous travaillez bien dans une région différente dans votre compte de gestion.

Au cours de la procédure de configuration, vous effectuez les tâches principales suivantes dans Quick Setup une fonctionnalité de AWS Systems Manager.

- Tâche 1 : enregistrer un compte administrateur délégué pour votre organisation

  Les tâches liées aux modifications effectuées en utilisant Change Manager sont gérées dans l'un de vos comptes membres spécifié comme compte d'administrateur délégué. Le compte d'administrateur délégué que vous enregistrez pour Change Manager devient le compte d'administrateur délégué pour toutes vos opérations Systems Manager. (Vous avez peut-être délégué des comptes d'administrateur pour d'autres Services AWS). Votre compte d'administrateur délégué pour Change Manager, qui est différent de votre compte de gestion, gère les activités de modification au sein de votre organisation, notamment les modèles de modifications, les demandes de modifications et les approbations qui s'y rapportent. Dans le compte d'administrateur délégué, vous spécifiez également d'autres options de configuration pour vos opérations Change Manager.

> **Important**
>
> Le compte d'administrateur délégué doit être le seul membre de l'unité d'organisation (UO) à laquelle il est affecté dans Organizations.

- Tâche 2 : définir et spécifier des politiques d'accès du runbook pour les rôles de demandeur de modification ou les fonctions professionnelles personnalisées que vous voulez utiliser pour vos Change Manageropérations

  Pour créer des demandes de modification dansChange Manager, les utilisateurs de vos comptes membres doivent disposer d'autorisations AWS Identity and Access Management (IAM) leur permettant d'accéder uniquement aux runbooks d'automatisation et aux modèles de modification que vous choisissez de mettre à leur disposition.

  > **Note**
  >
  > Lorsqu'un utilisateur crée une demande de modification, il sélectionne d'abord un modèle de modification. Avec ce modèle de modification, plusieurs runbooks peuvent être disponibles, mais l'utilisateur ne peut en sélectionner qu'un seul pour chaque demande de modification. Les modèles de modifications peuvent également être configurés pour autoriser les utilisateurs à inclure n'importe quel runbook disponible dans leurs demandes.

  Pour octroyer les autorisations nécessaires, Change Manager utilise le concept de fonctions professionnelles, qui est partagé par IAM. Cependant, contrairement aux politiques gérées par AWS pour les fonctions professionnelles dans IAM, ici, vous spécifiez les noms de vos fonctions professionnelles Change Manager et les autorisations IAM pour ces fonctions professionnelles.

  Lorsque vous configurez une fonction professionnelle, nous vous recommandons de créer une politique personnalisée et de ne fournir que les autorisations nécessaires pour effectuer des tâches de gestion des modifications. Par exemple, vous pouvez spécifier des autorisations limitant les utilisateurs à cet ensemble spécifique de runbooks selon des fonctions professionnelles définies.

  Par exemple, vous pouvez créer une fonction professionnelle avec le nom `DBAdmin`.
  Pour cette fonction professionnelle, vous pouvez octroyer uniquement les autorisations nécessaires pour les runbooks liés à des bases de données Amazon DynamoDB, comme `AWS-CreateDynamoDbBackup` et `AWSConfigRemediation-DeleteDynamoDbTable`.

Ou alors vous pouvez octroyer à certains utilisateurs uniquement les autorisations nécessaires pour utiliser les runbooks liés à des compartiments Amazon Simple Storage Service (Amazon S3), comme `AWS-ConfigureS3BucketLogging` et `AWSConfigRemediation-ConfigureS3BucketPublicAccessBlock`.

Le processus de configuration de Change Manager dans Quick Setup met également à votre disposition un ensemble d'autorisations administratives complètes de Systems Manager, que vous pouvez appliquer au rôle administratif que vous créez.

Chaque configuration de Quick Setup dans Change Manager déployée crée une fonction professionnelle dans votre compte d'administrateur délégué avec les autorisations d'exécuter des modèles Change Manager et des runbooks Automation dans les unités organisationnelles que vous avez sélectionnées. Vous pouvez créer jusqu'à 15 configurations Quick Setup pour Change Manager.

- Tâche 3 : choisir les comptes membres de votre organisation à utiliser avec Change Manager

  Vous pouvez utiliser Change Manager avec tous les comptes membres de toutes vos unités organisationnelles configurées dans Organizations, et dans toutes les Régions AWS où ils fonctionnent. Si vous préférez, vous pouvez utiliser Change Manager avec seulement quelques-unes de vos unités organisationnelles.

> **Important**
>
> Avant de commencer cette procédure, nous vous recommandons vivement de prendre connaissance des différentes étapes qui la composent, afin de comprendre les choix de configuration que vous effectuez et les autorisations que vous octroyez. En particulier, planifiez les fonctions professionnelles personnalisées que vous allez créer et les autorisations que vous affectez à chaque fonction professionnelle. De la sorte, lorsque vous attacherez les politiques de fonctions professionnelles créées à des utilisateurs individuels, des groupes d'utilisateurs ou des rôles IAM, ils ne recevront que les autorisations que vous entendez leur octroyer.
>
> Il est recommandé de commencer par configurer le compte d'administrateur délégué à l'aide de l'identifiant de connexion d'un Compte AWS administrateur. Ensuite, configurez les fonctions professionnelles et leurs autorisations après avoir créé des modèles de modifications et identifié les runbooks que chacun d'entre eux utilise.

Pour configurer Change Manager pour une utilisation avec une organisation,, exécutez la tâche suivante dans la zone Quick Setup de la console Systems Manager.

Répétez cette tâche pour chaque fonction professionnelle que vous voulez créer pour votre organisation. Chaque fonction professionnelle créée peut avoir des autorisations pour un ensemble différent d'unités organisationnelles.

Pour configurer une organisation pour Change Manager dans le compte de gestion Organizations

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Quick Setup.
3. Sur la carte Change Manager, choisissez Create (Créer).
4. Pour Delegated administrator account (Compte d'administrateur délégué), saisissez l'ID du Compte AWS que vous voulez utiliser pour gérer les modèles de modifications, les demandes de modifications et les flux de travail de runbook dans Change Manager.

   Si vous avez précédemment spécifié un compte d'administrateur délégué pour Systems Manager, son ID figure déjà dans ce champ.

   > **Important**
   >
   > Le compte d'administrateur délégué doit être le seul membre de l'unité d'organisation (UO) à laquelle il est affecté dans Organizations.
   > Si le compte d'administrateur délégué que vous enregistrez est désinscrit ultérieurement de ce rôle, le système supprime ses autorisations pour gérer les opérations du Systems Manager en même temps. N'oubliez pas que vous devrez revenir à Quick Setup, désigner un compte d'administrateur délégué différent et spécifier à nouveau toutes les fonctions professionnelles et les autorisations.
   > Si vous utilisez Change Manager au sein d'une organisation, nous vous recommandons de toujours apporter les modifications à partir du compte d'administrateur délégué. Bien qu'il soit possible d'apporter des modifications à partir d'autres comptes de l'organisation, celles-ci ne seront pas signalées ou affichées à partir du compte d'administrateur délégué.

5. Dans la section Autorisations de demander et d'apporter des modifications, procédez comme suit.

> **Note**
>
> Chaque configuration de déploiement que vous créez fournit la politique d'autorisations d'une seule fonction professionnelle. Vous pourrez revenir à Quick Setup ultérieurement pour créer d'autres fonctions professionnelles lorsque vous aurez créé des modèles de modifications à utiliser dans vos opérations.

Pour créer un rôle administratif : pour une fonction professionnelle d'administrateur qui dispose d'autorisations IAM pour toutes les actions AWS , procédez comme suit.

> **Important**
>
> L'octroi d'autorisations administratives complètes aux utilisateurs doit être effectué avec parcimonie, et uniquement si leurs rôles nécessitent un accès complet à Systems Manager. Pour obtenir des informations importantes sur les considérations de sécurité relatives à l'accès à Systems Manager, veuillez consulter Gestion des identités et des accès pour AWS Systems Manager et Bonnes pratiques de sécurité pour Systems Manager.

1. Pour Job function (Fonction professionnelle), saisissez un nom pour identifier ce rôle et ses autorisations, **`My AWS Admin`** par exemple.
2. Pour Role and permissions option (Option de rôle et d'autorisations), sélectionnez Autorisations d'administrateur.

Pour créer d'autres fonctions professionnelles : pour créer un rôle non administratif, procédez comme suit :

1. Pour Job function (Fonction professionnelle), saisissez un nom pour identifier ce rôle et suggérer ses autorisations. Le nom que vous sélectionnez doit représenter la portée des runbooks pour lesquels vous fournirez des autorisations, `DBAdmin` ou `S3Admin` par exemple.
2. Pour Role and permissions option (Option de rôle et d'autorisations), sélectionnez Custom persmissions (Autorisations personnalisées).
3. Dans l'éditeur de politique d'autorisations, saisissez les autorisations IAM, au format JSON, à octroyer à cette fonction professionnelle.

> **Tip**
>
> Nous vous recommandons d'utiliser l'éditeur de politique IAM pour construire votre politique, puis de coller le JSON de la politique dans le champ Politique d'autorisations.

Exemple de politique : gestion de la base de données DynamoDB

Par exemple, vous pouvez commencer par le contenu de la politique qui fournit des autorisations pour travailler avec les documents Systems Manager (documents SSM) auxquels la fonction professionnelle doit accéder. Voici un exemple de contenu de politique qui donne accès à tous les runbooks d'automatisation AWS gérés liés aux bases de données DynamoDB et à deux modèles de modification créés dans l' Compte AWS `123456789012`exemple, dans la région USA Est (Ohio) (). `us-east-2`

La politique inclut également l'autorisation pour l'opération StartChangeRequestExecution, qui est obligatoire pour créer une demande de modification dans Change Calendar.

> **Note**
>
> Cet exemple n'est pas exhaustif. Des autorisations supplémentaires peuvent être nécessaires pour travailler avec d'autres AWS ressources, telles que des bases de données et des nœuds.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:CreateDocument",
                "ssm:DescribeDocument",
                "ssm:DescribeDocumentParameters",
                "ssm:DescribeDocumentPermission",
                "ssm:GetDocument",
                "ssm:ListDocumentVersions",
                "ssm:ModifyDocumentPermission",
```

```
                "ssm:UpdateDocument",
                "ssm:UpdateDocumentDefaultVersion"
            ],
            "Resource": [
                "arn:aws:ssm:region:*:document/AWS-CreateDynamoDbBackup",
                "arn:aws:ssm:region:*:document/AWS-AWS-DeleteDynamoDbBackup",
                "arn:aws:ssm:region:*:document/AWS-DeleteDynamoDbTableBackups",
                "arn:aws:ssm:region:*:document/AWSConfigRemediation-
DeleteDynamoDbTable",
                "arn:aws:ssm:region:*:document/AWSConfigRemediation-
EnableEncryptionOnDynamoDbTable",
                "arn:aws:ssm:region:*:document/AWSConfigRemediation-
EnablePITRForDynamoDbTable",
                "arn:aws:ssm:region:123456789012:document/MyFirstDBChangeTemplate",
                "arn:aws:ssm:region:123456789012:document/MySecondDBChangeTemplate"
            ]
        },
        {
            "Effect": "Allow",
            "Action": "ssm:ListDocuments",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "ssm:StartChangeRequestExecution",
            "Resource": "arn:aws:ssm:region:123456789012:automation-definition/*:*"
        }
    ]
}
```

Pour de plus amples informations sur les politiques IAM, veuillez consulter Gestion des accès pour des ressources AWS et Création de politiques IAM dans le Guide de l'utilisateur IAM.

6. Dans la section Cibles, choisissez d'octroyer des autorisations pour la fonction professionnelle que vous créez à l'ensemble de votre organisation ou à certaines de vos unités organisationnelles uniquement.

   Si vous sélectionnez Ensemble de l'organisation, passez à l'étape 9.

   Si vous sélectionnez Custom (Personnalisé), passez à l'étape 8.

7. Dans la section UO cibles, sélectionnez les cases des unités organisationnelles à utiliser avec Change Manager.

8. Sélectionnez Create (Créer).

Une fois que le système a terminé de configurer Change Managerpour votre organisation, il affiche un résumé de vos déploiements. Ces informations récapitulatives incluent le nom du rôle créé pour la fonction professionnelle que vous avez configurée. Par exemple, `AWS-QuickSetup-SSMChangeMgr-DBAdminInvocationRole`.

> **Note**
>
> Quick Setuputilise AWS CloudFormation StackSets pour déployer vos configurations. Vous pouvez également afficher des informations sur une configuration de déploiement terminée dans la console AWS CloudFormation . Pour plus d'informations à ce sujet StackSets, consultez la section Travailler avec AWS CloudFormation StackSets dans le guide de AWS CloudFormation l'utilisateur.

Dans l'étape suivante, vous allez configurer des options Change Manager supplémentaires. Vous pouvez effectuer cette tâche dans votre compte d'administrateur délégué ou dans n'importe quel compte d'une unité d'organisation dont vous avez autorisé l'utilisation avec Change Manager. Parmi les options que vous configurez, vous pouvez choisir une option de gestion des identités utilisateur, spécifier les utilisateurs qui peuvent vérifier et approuver ou rejeter les modèles de modifications et les demandes de modifications, et choisir les options de bonnes pratiques à autoriser pour votre organisation. Pour plus d'informations, veuillez consulter Configuration d'options et de bonnes pratiques Change Manager.

Configuration d'options et de bonnes pratiques Change Manager

Les tâches décrites dans cette section doivent être effectuéesChange Manager, que vous utilisiez une fonctionnalité au sein d' AWS Systems Manager une organisation ou au sein d'une seule organisation Compte AWS.

Si vous utilisez Change Manager pour une organisation, vous pouvez effectuer les tâches suivantes dans votre compte d'administrateur délégué ou dans n'importe quel compte d'une unité d'organisation dont vous avez autorisé l'utilisation avec Change Manager.

Rubriques

- Tâche 1 : configuration de vérificateurs de gestion des identités des utilisateurs et de modèles Change Manager

- Tâche 2 : configuration d'approbateurs d'événements de gel des modifications et de bonnes pratiques Change Manager
- Configuration des rubriques Amazon SNS pour les notifications Change Manager

Tâche 1 : configuration de vérificateurs de gestion des identités des utilisateurs et de modèles Change Manager

Effectuez la tâche de cette procédure la première fois que vous accédez à Change Manager. Vous pouvez mettre à jour ces paramètres de configuration ultérieurement en revenant à Change Manager et en choisissant Modifier sous l'onglet Settings (Paramètres).

Pour configurer des vérificateurs de gestion des identités des utilisateurs et de modèles Change Manager

1. Connectez-vous au AWS Management Console.

   Si vous utilisez Change Manager pour une organisation, connectez-vous avec les informations d'identification de votre compte d'administrateur délégué. L'utilisateur doit disposer des autorisations AWS Identity and Access Management (IAM) nécessaires pour effectuer des mises à jour de vos paramètres Change Manager.
2. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
3. Dans le panneau de navigation, sélectionnez Change Manager.
4. Sur la page d'accueil du service, effectuez l'une des actions suivantes en fonction des options disponibles :

   - Si vous utilisez Change Manager avec AWS Organizations , choisissez Configurer un compte délégué.
   - Si vous l'utilisez Change Manager avec un seul Compte AWS appareil, choisissez Configurer Change Manager.

     -ou-

     Sélectionnez Créer un exemple de demande de modification, Ignore, puis sélectionnez l'onglet Settings (Paramètres).
5. Pour Gestion des identités utilisateur, sélectionnez l'une des options suivantes.

- AWS Identity and Access Management (IAM) — Identifiez les utilisateurs qui font et approuvent les demandes et effectuent d'autres actions en Change Manager utilisant vos utilisateurs, groupes et rôles existants.
- AWS IAM Identity Center (IAM Identity Center) — Autorisez IAM Identity Center à créer et à gérer des identités, ou à vous connecter à votre source d'identité existante pour identifier les utilisateurs qui effectuent des actions dans. Change Manager

6. Dans la section Template reviewer notification (Notification du vérificateur de modèle), spécifiez les rubriques Amazon Simple Notification Service (Amazon SNS) à utiliser afin d'informer les vérificateurs de modèles qu'il est possible de vérifier un nouveau modèle de modification ou une nouvelle version d'un modèle de modification. Vérifiez que la rubrique Amazon SNS que vous sélectionnez est bien configurée pour envoyer des notifications à vos vérificateurs de modèles.

   Pour obtenir des informations sur la création et la configuration de rubriques Amazon SNS pour les notifications des vérificateurs de modèles de modifications, consultez Configuration des rubriques Amazon SNS pour les notifications Change Manager.

   1. Pour spécifier la rubrique Amazon SNS pour les notifications des vérificateurs de modèles, sélectionnez l'une des options suivantes :
      - Saisir un Amazon Resource Name (ARN) SNS : pour ARN de rubrique, saisissez l'ARN d'une rubrique Amazon SNS existante. Cette rubrique peut se trouver dans n'importe quel compte de votre organisation.
      - Sélectionner une rubrique SNS existante : pour Rubrique de notification cible, sélectionnez l'ARN d'une rubrique Amazon SNS existante dans votre Compte AWS actuel. (Cette option n'est pas disponible si vous n'avez pas encore créé de rubrique Amazon SNS dans votre Compte AWS et Région AWS.)

      > **Note**
      >
      > La rubrique Amazon SNS que vous sélectionnez doit être configurée pour spécifier les notifications qu'elle envoie, ainsi que les abonnés auxquels elles sont envoyées. Sa politique d'accès doit également octroyer des autorisations à Systems Manager de sorte que Change Manager puisse envoyer des notifications. Pour plus d'informations, consultez Configuration des rubriques Amazon SNS pour les notifications Change Manager.

   2. Sélectionnez Ajouter une notification.

7. Dans la section Vérificateurs des modèles de modifications, sélectionnez les utilisateurs de votre organisation ou de votre compte qui vérifieront les nouveaux modèles de modifications ou modifieront les versions de modèles avant qu'ils soient utilisés dans vos opérations.

   Les vérificateurs de modèles de modifications sont chargés de vérifier l'adéquation et la sécurité des modèles que d'autres utilisateurs ont envoyés en vue d'une utilisation dans des flux de travail de runbook Change Manager.

   Sélectionnez les vérificateurs des modèles de modifications de la façon suivante :

   1. Choisissez Ajouter.
   2. Cochez la case en regard du nom de chaque utilisateur, groupe ou rôle IAM que vous souhaitez affecter en tant que vérificateur de modèle de modification.
   3. Sélectionnez Ajouter des approbateurs.
8. Sélectionnez Submit (Envoyer).

Une fois ce processus de configuration initiale terminé, configurez des paramètres et des bonnes pratiques Change Manageren suivant les étapes décrites dans Tâche 2 : configuration d'approbateurs d'événements de gel des modifications et de bonnes pratiques Change Manager.

Tâche 2 : configuration d'approbateurs d'événements de gel des modifications et de bonnes pratiques Change Manager

Après avoir effectué les étapes de Tâche 1 : configuration de vérificateurs de gestion des identités des utilisateurs et de modèles Change Manager, vous pouvez désigner des vérificateurs supplémentaires pour les demandes de modifications lors d'événements de gel des modifications et spécifier les bonnes pratiques disponibles que vous souhaitez autoriser pour vos opérations Change Manager.

Un événement de gel des modifications signifie que des restrictions sont en place dans le calendrier des modifications actuel (c'est l'état du calendrier qui s'y AWS Systems Manager Change Calendar trouve`CLOSED`). Dans ce cas, outre les approbateurs réguliers des demandes de modifications, ou si la demande de modification est créée à l'aide d'un modèle qui autorise les approbations automatiques, les approbateurs de gel des modifications doivent octroyer l'autorisation d'exécuter cette demande de modification. Si ce n'est pas le cas, la modification ne sera pas traitée tant que le calendrier ne retrouvera pas le statut `OPEN`.

Pour configurer d'approbateurs d'événements de gel des modifications et de bonnes pratiques Change Manager

1. Dans le panneau de navigation, sélectionnez Change Manager.
2. Sélectionnez l'onglet Settings (Paramètres), puis Edit (Modifier).
3. Dans Approbateurs d'événements de gel des modifications, sélectionnez les utilisateurs de votre organisation ou de votre compte qui peuvent approuver les modifications à exécuter même lorsque le calendrier utilisé dans Change Calendar a le statut FERMÉ.

   > **Note**
   >
   > Pour autoriser les vérifications de gel des modifications, vous devez activer l'option Vérifier les événements de modifications restreintes dans le calendrier des modifications dans les Bonnes pratiques.

   Sélectionnez les approbateurs d'événements de gel des modifications en procédant comme suit :

   1. Choisissez Ajouter.
   2. Cochez la case en regard du nom de chaque utilisateur, groupe ou rôle IAM que vous souhaitez affecter en tant qu'approbateur pour les événements de gel des modifications.
   3. Sélectionnez Ajouter des approbateurs.
4. Dans la section Bonnes pratiques, en bas de la page, activez les bonnes pratiques que vous souhaitez appliquer pour chacune des options suivantes.

   - Option : Vérifier les événements de modifications restreintes dans le calendrier des modifications

     Pour spécifier que Change Manager vérifie un calendrier dans Change Calendar pour s'assurer que les modifications ne sont pas bloquées par des événements planifiés, cochez d'abord la case Activé, puis sélectionnez le calendrier dans lequel vérifier les événements de modifications restreintes dans la liste Calendrier des modifications.

     Pour plus d'informations sur Change Calendar, consultez AWS Systems Manager Change Calendar.
   - Option :Rubrique SNS pour les approbateurs d'événements fermés

1. Sélectionnez l'une des options suivantes pour spécifier la rubrique Amazon Simple Notification Service (Amazon SNS) de votre compte à utiliser pour envoyer des notifications aux approbateurs lors d'événements de gel des modifications. (Notez que vous devez également spécifier des approbateurs dans la section Approbateurs d'événements de gel des modifications au-dessus des Bonnes pratiques.)
   - Saisir un Amazon Resource Name (ARN) SNS : pour ARN de rubrique, saisissez l'ARN d'une rubrique Amazon SNS existante. Cette rubrique peut se trouver dans n'importe quel compte de votre organisation.
   - Sélectionner une rubrique SNS existante : pour Rubrique de notification cible, sélectionnez l'ARN d'une rubrique Amazon SNS existante dans votre Compte AWS actuel. (Cette option n'est pas disponible si vous n'avez pas encore créé de rubrique Amazon SNS dans votre Compte AWS et Région AWS.)

   > **Note**
   >
   > La rubrique Amazon SNS que vous sélectionnez doit être configurée pour spécifier les notifications qu'elle envoie, ainsi que les abonnés auxquels elles sont envoyées. Sa politique d'accès doit également octroyer des autorisations à Systems Manager de sorte que Change Manager puisse envoyer des notifications. Pour plus d'informations, consultez Configuration des rubriques Amazon SNS pour les notifications Change Manager.

2. Sélectionnez Ajouter une notification.

- Option : Exiger des surveillances pour tous les modèles

  Si vous voulez vous assurer que tous les modèles de votre organisation ou de votre compte contiennent une CloudWatch alarme Amazon pour surveiller votre opération de modification, cochez la case Activé.

- Option : Exiger la vérification et l'approbation du modèle avant son utilisation

  Pour vous assurer qu'aucune demande de modification n'est créée et qu'aucun flux de travail de runbook n'est exécuté sans être basés sur un modèle qui a été vérifié et approuvé, cochez la case Activé.

5. Sélectionnez Enregistrer.

Configuration des rubriques Amazon SNS pour les notifications Change Manager

Vous pouvez configurer Change Manager, une fonctionnalité de AWS Systems Manager, pour qu'il envoie des notifications à une rubrique Amazon Simple Notification Service (Amazon SNS) à propos d'événements liés à des demandes de modifications et des modèles de modifications. Effectuez les tâches suivantes pour recevoir des notifications pour les événements Change Manager auxquels vous ajoutez une rubrique.

Rubriques

- Tâche 1 : Créer une rubrique Amazon SNS et s'y abonner
- Tâche 2 : Mise à jour de la politique d'accès Amazon SNS
- Tâche 3 : (facultatif) mettre à jour la politique d'accès AWS Key Management Service

Tâche 1 : Créer une rubrique Amazon SNS et s'y abonner

Pour commencer, vous devez créer une rubrique Amazon SNS à laquelle vous vous abonnez. Pour plus d'informations, consultez Création d'une rubrique Amazon SNS et Abonnement à une rubrique Amazon SNS dans le Guide du développeur Amazon Simple Notification Service.

> **Note**
>
> Pour recevoir des notifications, vous devez spécifier l'Amazon Resource Name (ARN) d'une rubrique Amazon SNS qui se trouve dans la même Région AWS et le même Compte AWS que le compte administrateur délégué.

Tâche 2 : Mise à jour de la politique d'accès Amazon SNS

Utilisez la procédure suivante pour mettre à jour la politique d'accès Amazon SNS afin que Systems Manager puisse publier les notifications Change Manager dans la rubrique Amazon SNS créée dans la tâche 1. Si cette tâche n'est pas effectuée ,Change Manager n'a pas l'autorisation d'envoyer des notifications à propos d'événements auxquels vous ajoutez la rubrique.

1. Connectez-vous à 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 Topics (Rubriques).
3. Sélectionnez la rubrique que vous avez créée dans la tâche 1, puis sélectionnez Edit (Modifier).

4. Développez la politique d'accès.
5. Ajoutez et mettez à jour le bloc `Sid` suivant dans la politique existante, et remplacez chaque *`espace réservé aux entrées utilisateur`* par vos propres informations.

```
{
    "Sid": "Allow Change Manager to publish to this topic",
    "Effect": "Allow",
    "Principal": {
        "Service": "ssm.amazonaws.com"
    },
    "Action": "sns:Publish",
    "Resource": "arn:aws:sns:region:account-id:topic-name",
    "Condition": {
        "StringEquals": {
            "aws:SourceAccount": [
                "account-id"
            ]
        }
    }
}
```

   Saisissez ce bloc après le bloc `Sid` existant et remplacez *`region`*, *`account-id`* (ID de compte) et *`topic-name`* (nom de rubrique) par les valeurs appropriées pour la rubrique que vous avez créée.
6. Sélectionnez Enregistrer les modifications.

Le système enverra maintenant des notifications à la rubrique Amazon SNS lorsque le type d'événement que vous ajoutez à la rubrique se produira.

> **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.

Tâche 3 : (facultatif) mettre à jour la politique d'accès AWS Key Management Service

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 politique d'accès de la AWS

KMS key que vous avez choisie lors de la configuration de la rubrique. Utilisez la procédure suivante pour mettre à jour la politique d'accès afin que Systems Manager puisse publier les notifications d'approbation Change Manager dans la rubrique Amazon SNS créée dans la tâche 1.

1. Ouvrez la console AWS KMS à l'adresse https://console.aws.amazon.com/kms.
2. Dans le volet de navigation, sélectionnez Clés gérées par le client.
3. Sélectionnez l'ID de la clé gérée par le client que vous avez choisie lors de la création de la rubrique.
4. Dans la section Key policy (Politique de clé), sélectionnez Switch to policy view (Passer à la vue de politique).
5. Sélectionnez Edit (Modifier).
6. Entrez le bloc `Sid` suivant après l'un des blocs `Sid` existants dans la politique existante. Remplacez chaque *espace réservé à la saisie de l'utilisateur* par vos propres informations.

```
{
    "Sid": "Allow Change Manager to decrypt the key",
    "Effect": "Allow",
    "Principal": {
        "Service": "ssm.amazonaws.com"
    },
    "Action": [
        "kms:Decrypt",
        "kms:GenerateDataKey*"
    ],
    "Resource": "arn:aws:kms:region:account-id:key/key-id",
    "Condition": {
        "StringEquals": {
            "aws:SourceAccount": [
                "account-id"
            ]
        }
    }
}
```

7. Maintenant, entrez le bloc `Sid` suivant après l'un des blocs `Sid` existants dans la politique de ressources pour aider à prévenir le problème du député confus entre services.

Ce bloc utilise les clés de contexte de condition globale aws:SourceArn et aws:SourceAccount pour limiter les autorisations que Systems Manager accorde à un autre service pour la ressource.

Remplacez chaque *espace réservé à la saisie de l'utilisateur* par vos propres informations.

```
{
  "Version": "2008-10-17",
  "Statement": [
    {
      "Sid": "Configure confused deputy protection for AWS KMS keys used in Amazon
 SNS topic when called from Systems Manager",
      "Effect": "Allow",
      "Principal": {
        "Service": "ssm.amazonaws.com"
      },
      "Action": [
        "sns:Publish"
      ],
      "Resource": "arn:aws:sns:region:account-id:topic-name",
      "Condition": {
        "ArnLike": {
          "aws:SourceArn": "arn:aws:ssm:region:account-id:*"
        },
        "StringEquals": {
          "aws:SourceAccount": "account-id"
        }
      }
    }
  ]
}
```

8. Sélectionnez Save Changes (Enregistrer les modifications).

Configuration des rôles et des autorisations pour Change Manager

Par défaut, Change Manager ne dispose pas d'autorisation pour effectuer des actions sur vos instances. Vous devez accorder l'accès en utilisant un rôle de service AWS Identity and Access Management (IAM) ou en assumant un rôle. Ce rôle permet à Change Manager d'exécuter en toute sécurité les flux de travail du Runbook spécifiés dans une demande de modification approuvée

en votre nom. Le rôle accorde la AssumeRoleconfiance à AWS Security Token Service (AWS STS)Change Manager.

En accordant ces autorisations à un rôle pour agir au nom des utilisateurs d'une organisation, les utilisateurs n'ont pas besoin de se voir accorder eux-mêmes ce tableau d'autorisations. Les actions permises par les autorisations se limitent uniquement aux opérations approuvées.

Lorsque les utilisateurs de votre compte ou de votre organisation créent une demande de modification, ceux-ci peuvent sélectionner ce rôle de responsable pour effectuer les opérations de modification.

Vous pouvez créer un nouveau rôle de responsable pour Change Manager ou mettre à jour un rôle existant avec les autorisations nécessaires.

Si vous avez besoin de créer une fonction du service pour Change Manager, exécutez les tâches suivantes.

Tâches

- Tâche 1 : Création d'une politique de rôle de responsable pour Change Manager
- Tâche 2 : Création d'une politique de rôle de responsable pour Change Manager
- Tâche 3 : Attacher la politique iam:PassRole à d'autres rôles
- Tâche 4 : Ajouter des politiques intégrées à un rôle d'assume pour invoquer d'autres Services AWS
- Tâche 5 : Configuration de l'accès utilisateur à Change Manager

Tâche 1 : Création d'une politique de rôle de responsable pour Change Manager

Suivez la procédure suivante pour créer la politique que vous attacherez à votre rôle de responsable Change Manager.

Pour créer une politique de rôle de responsabilité pour Change Manager

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le volet de navigation, choisissez Policies, puis Create Policy.
3. Dans la page Create policy (Créer une politique), choisissez l'onglet JSON et remplacez le contenu par défaut par le contenu suivant que vous modifierez pour vos opérations Change Manager au bout des étapes suivantes.

> **Note**
>
> Si vous créez une politique à utiliser avec un seul compte Compte AWS, et non avec une organisation possédant plusieurs comptes Régions AWS, vous pouvez omettre le premier bloc de relevés. L'autorisation `iam:PassRole` n'est pas requise pour le cas d'un compte unique utilisant Change Manager.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::delegated-admin-account-id:role/AWS-
SystemsManager-job-functionAdministrationRole",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": "ssm.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeDocument",
                "ssm:GetDocument",
                "ssm:StartChangeRequestExecution"
            ],
            "Resource": [
                "arn:aws:ssm:region:account-id:automation-definition/template-name:
$DEFAULT",
                "arn:aws:ssm:region::document/template-name"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:ListOpsItemEvents",
                "ssm:GetOpsItem",
                "ssm:ListDocuments",
```

```
                "ssm:DescribeOpsItems"
            ],
            "Resource": "*"
        }
    ]
}
```

4. Pour l'action `iam:PassRole`, mettez à jour la `Resource` afin d'inclure les ARN de toutes les fonctions de travail définies pour votre organisation auxquelles vous souhaitez accorder des autorisations pour introduire des flux de travail Runbook.
5. Remplacez les espaces réservés *`région`*, *`id de compte`*, *`nom du modèle`*, *`ID de compte d'administrateur délégué`*, et *`fonction de travail`* espaces réservés par des valeurs pour vos opérations Change Manager.
6. Pour la seconde `Resource`, modifiez la liste pour inclure tous les modèles de modification pour lesquels vous souhaitez accorder des autorisations. Sinon, indiquez `"Resource": "*"` pour accorder des autorisations pour tous les modèles de modification de votre organisation.
7. Choisissez Suivant : Balises.
8. (Facultatif) Ajoutez une ou plusieurs paires clé-valeur d'identification afin d'organiser, de suivre ou de contrôler l'accès pour cette politique.
9. Choisissez Suivant : vérification.
10. Dans la page Review policy (Vérification de la politique), saisissez un nom dans la zone Name (Nom), tel que **`MyChangeManagerAssumeRole`**, puis saisissez une description facultative.
11. Choisissez Create policy (Créer une politique), et continuez vers Tâche 2 : Création d'une politique de rôle de responsable pour Change Manager.

Tâche 2 : Création d'une politique de rôle de responsable pour Change Manager

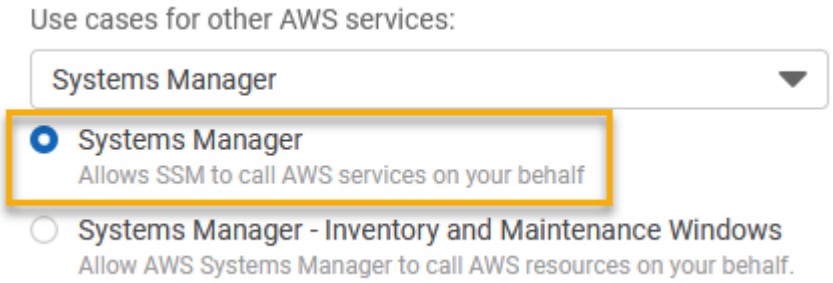
Suivez la procédure suivante pour créer un rôle de responsable Change Manager, une sorte de fonction du service pour Change Manager.

Pour créer un rôle d'assumer pour Change Manager

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le volet de navigation, sélectionnez Rôles, puis Créer un rôle.
3. Pour Select trusted entity (Sélectionner une entité de confiance), effectuez les choix suivants :

    1. Pour Type d'entité de confiance, choisissez Service AWS

2. Pour les cas d'utilisation pour les autres Services AWS, choisissez Systems Manager
3. Choisissez Systems Manager, comme illustré dans l'image suivante.



4. Choisissez Suivant.
5. Dans la page Attached permissions policy (¨Politique d'autorisations attachées), recherchez la politique de rôle de responsable que vous avez créée dansTâche 1 : Création d'une politique de rôle de responsable pour Change Manager, comme **`MyChangeManagerAssumeRole`**.
6. Cochez la case cà côté du nom de la politique de rôle de responsable, puis choisissez Next: Tags (Suivant : balises).
7. Pour Role name (Nom du rôle), saisissez un nom pour votre nouveau profil d'instance, par exemple **`MyChangeManagerAssumeRole`**.
8. (Facultatif) Pour Description, saisissez une description pour ce role d'instance.
9. (Facultatif) Ajoutez une ou plusieurs paires clé-valeur d'identification afin d'organiser, de suivre ou de contrôler l'accès pour ce rôle.
10. Choisissez Suivant : vérification.
11. (Facultatif) Pour Tags (Balises), ajoutez une ou plusieurs paires clé-valeur d'identification afin d'organiser, de suivre ou de contrôler l'accès pour ce rôle, puis sélectionnez Create role (Créer le rôle). Le système vous renvoie à la page Rôles.
12. Sélectionnez Créer un rôle. Le système vous renvoie à la page Rôles.
13. Sur la page Rôles, sélectionnez le rôle que vous venez de créer pour ouvrir la page Récapitulatif.

Tâche 3 : Attacher la politique **`iam:PassRole`** à d'autres rôles

Procédez comme suit pour attacher la politique `iam:PassRole` à un profil d'instance IAM ou à une fonction du service IAM. (Le service Systems Manager utilise des profils d'instance IAM pour communiquer avec les instances EC2. Pour les nœuds gérés non EC2 dans un environnement hybride et multicloud, une fonction du service IAM est utilisée à la place.)

En attachant la politique `iam:PassRole`, le service Change Manager peut transmettre des autorisations de rôle à d'autres services ou fonctionnalités Systems Manager lors de l'exécution des flux de travail de runbook.

Pour attacher la politique **`iam:PassRole`** à un profil d'instance ou une fonction du service IAM

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation, sélectionnez Rôles.
3. Recherchez le rôle de responsable Change Manager que vous avez créé, comme **`MyChangeManagerAssumeRole`** et définissez son nom.
4. Dans la page Summary (Résumé) du rôle de responsable, sélectionnez l'onglet Autorisations.
5. Choisissez Add permissions, Create inline policy (Ajouter des autorisations, Créer une politique en ligne).
6. Dans la page Créer une politique, sélectionnez l'onglet Éditeur visuel.
7. Sélectionnez Service, puis sélectionnez IAM.
8. Dans la zone de texte Actions de filtrage**`PassRole`**, entrez, puis choisissez l'PassRoleoption.
9. Développer les Ressources. Vérifiez que Spécifique est sélectionné, puis sélectionnez Add ARN (Ajouter l'ARN).
10. Dans le champ Specify ARN for role (Spécifier l'ARN du rôle), saisissez l'ARN du rôle de profil d'instance IAM ou celui de la fonction du service IAM auquel vous souhaitez transmettre les autorisations de rôle de responsable. Le système remplit automatiquement les champs Compte et Role name with path (Nom du rôle avec chemin d'accès).
11. Choisissez Ajouter.
12. Sélectionnez Review policy (Examiner une politique).
13. Pour Name (Nom), entrez un nom pour identifier cette politique, puis choisissez Create policy (Créer une politique).

Plus d'informations

- Configurer les autorisations d'instance requises pour Systems Manager
- Créez le rôle de service IAM requis pour Systems Manager dans les environnements hybrides et multicloud

Tâche 4 : Ajouter des politiques intégrées à un rôle d'assume pour invoquer d'autres Services AWS

Lorsqu'une demande de modification invoque d'autres services en Services AWS utilisant le rôle d'Change Managerassumer, le rôle d'assumer doit être configuré avec l'autorisation d'invoquer ces services. Cette exigence s'applique à tous les runbooks AWS Automation (runbooks AWS-*) susceptibles d'être utilisés dans une demande de modification, tels que les runbooks et les `AWS-ConfigureS3BucketLogging` runbooks. `AWS-CreateDynamoDBBackup` `AWS-RestartEC2Instance` Cette exigence s'applique également à tous les runbooks personnalisés que vous créez et qui invoquent d'autres services Services AWS en utilisant des actions qui appellent d'autres services. Par exemple, si vous exécutez les actions `aws:executeAwsApi`, `aws:CreateStack` ou `aws:copyImage`, vous devez configurer la fonction du service avec l'autorisation de faire appel à ces services. Vous pouvez accorder des autorisations à d'autres Services AWS en ajoutant une politique IAM en ligne au rôle.

Pour ajouter une politique en ligne à une fonction assumée afin d'appeler à d'autres Services AWS (console IAM)

1. Connectez-vous à la console IAM AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation, choisissez Roles (Rôles).
3. Dans la liste, définissez le nom du rôle de responsable que vous souhaitez mettre à jour, par exemple `MyChangeManagerAssumeRole`.
4. Choisissez l'onglet Permissions (Autorisations).
5. Choisissez Add permissions, Create inline policy (Ajouter des autorisations, Créer une politique en ligne).
6. Sélectionnez l'onglet JSON.
7. Entrez un document de politique JSON pour le que Services AWS vous souhaitez invoquer. 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":"*"
      }
   ]
}
```

Afin d'obtenir des informations approfondies sur la terminologie IAM, consultez la Référence de politique JSON IAM dans le Guide de l'utilisateur IAM.

8. Lorsque vous avez terminé, sélectionnez Review policy (Examiner une politique). Le programme de validation de politique signale les éventuelles erreurs de syntaxe.
9. Pour Name (Nom), saisissez un nom pour identifier la politique que vous créez. Vérifiez le récapitulatif de politique pour voir les autorisations accordées par votre politique. Sélectionnez ensuite Créer une politique pour enregistrer votre travail.
10. Une fois que vous avez créé une politique en ligne, elle est automatiquement intégrée à votre rôle.

Tâche 5 : Configuration de l'accès utilisateur à Change Manager

Si votre utilisateur, votre groupe ou votre rôle dispose des autorisations d'administrateur, vous avez accès à Change Manager. Si vous ne disposez pas des autorisations d'administrateur, alors un administrateur doit vous assigner la politique gérée `AmazonSSMFullAccess` ou une politique octroyant des d'autorisations similaires à votre utilisateur, groupe ou rôle.

Suivez la procédure suivante afin de configurer un utilisateur pour utiliser Change Manager. L'utilisateur que vous sélectionnez aura l'autorisation de configurer et d'exécuter Change Manager.

En fonction de l'application d'identité que vous utilisez dans votre organisation, vous pouvez sélectionner l'une des trois options disponibles pour configurer l'accès des utilisateurs. Lors de la configuration de l'accès utilisateur, attribuez ou ajoutez les éléments suivants :

1. Attribuez la politique `AmazonSSMFullAccess` ou une politique comparable qui autorise l'accès à Systems Manager.
2. Attribuez la politique `iam:PassRole`.
3. Ajoutez l'ARN du rôle de responsable Change Manager que vous avez copié à la fin de Tâche 2 : Création d'une politique de rôle de responsable pour Change Manager.

Pour activer l'accès, ajoutez des autorisations à vos utilisateurs, groupes ou rôles :

- Utilisateurs et groupes dans AWS IAM Identity Center :

  Créez un jeu d'autorisations. Suivez les instructions de la rubrique Création d'un jeu d'autorisations du Guide de l'utilisateur AWS IAM Identity Center .

- Utilisateurs gérés dans IAM par un fournisseur d'identité :

  Créez un rôle pour la fédération d'identité. Pour plus d'informations, voir la rubrique Création d'un rôle pour un fournisseur d'identité tiers (fédération) du Guide de l'utilisateur IAM.

- Utilisateurs IAM :
  - Créez un rôle que votre utilisateur peut assumer. Suivez les instructions de la rubrique Création d'un rôle pour un utilisateur IAM du Guide de l'utilisateur IAM.
  - (Non recommandé) Attachez une politique directement à un utilisateur ou ajoutez un utilisateur à un groupe d'utilisateurs. Suivez les instructions de la rubrique Ajout d'autorisations à un utilisateur (console) du Guide de l'utilisateur IAM.

Vous en avez terminer avec la configuration des rôles requis pour Change Manager. Vous pouvez désormais utiliser l'ARN du rôle de responsabilité Change Manager dans vos opérations Change Manager.

Contrôler l'accès aux flux de travail de runbook d'approbation automatique

Dans chaque modèle de modification créé pour votre organisation ou votre compte, vous pouvez spécifier si les demandes de modifications créées à partir de ce modèle peuvent être exécutées en tant que des demandes de modifications approuvées automatiquement, ce qui signifie qu'elles s'exécutent automatiquement sans l'étape de vérification (à l'exception des événements de gel des modifications).

Toutefois, vous pouvez empêcher certains utilisateurs, groupes ou rôles AWS Identity and Access Management (IAM) d'exécuter des demandes de modifications approuvées automatiquement même si un modèle de modification l'autorise. Pour cela, vous pouvez utiliser la clé de condition `ssm:AutoApprove` pour l'opération `StartChangeRequestExecution` dans une politique IAM affectée à l'utilisateur, au groupe ou au rôle IAM.

Vous pouvez ajouter la politique suivante en tant que politique intégrée, dans laquelle la condition est spécifiée comme `false`, pour empêcher les utilisateurs d'exécuter des demandes de modifications à approbation automatique.

```
{
    "Version": "2012-10-17",
    "Statement": [
            {
            "Effect": "Allow",
            "Action": "ssm:StartChangeRequestExecution",
            "Resource": "*",
            "Condition": {
                "BoolIfExists": {
                    "ssm:AutoApprove": "false"
                }
            }
        }
    ]
}
```

Pour obtenir des informations sur la spécification de politiques intégrées, veuillez consulter Politiques en ligne et Ajout et suppression d'autorisations basées sur l'identité IAM dans le Guide de l'utilisateur IAM.

Pour de plus amples informations sur les clés de condition pour les politiques Systems Manager, veuillez consulter la rubrique Clés de condition pour Systems Manager.

Utilisation des Change Manager

Avec Change Manager, une fonctionnalité de AWS Systems Manager, des utilisateurs au sein de votre organisation ou dans un Compte AWS unique peuvent effectuer des tâches liées aux modifications pour lesquelles les autorisations nécessaires leur ont été octroyées. Les tâches Change Managercomprennent notamment :

- La création, la vérification, l'approbation ou le rejet de modèles de modifications.

  Un modèle de modification est un ensemble de paramètres de configuration dans Change Manager, qui définissent des éléments tels que les approbations requises, les runbooks disponibles et les options de notification relatives aux demandes de modifications.
- La création, la vérification, l'approbation ou le rejet de demandes de modifications.

  Une demande de modification est une demande d'exécution d'un runbook Automation dans Change Manager, qui met à jour une ou plusieurs ressources dans votre environnement AWS ou sur site. Une demande de modification est créée à l'aide d'un modèle de modification.
- Spécifiez les utilisateurs de votre organisation ou de votre compte qui peuvent devenir des vérificateurs de modèles de modifications et de demandes de modifications.
- Modifiez les paramètres de configuration, notamment la façon dont les identités utilisateur sont gérées dans Change Manager et les bonnes pratiques qui sont appliquées dans vos opérations Change Manager. Pour de plus amples informations sur la configuration des paramètres, veuillez consulter Configuration d'options et de bonnes pratiques Change Manager.

Rubriques

Utilisation des modèles de modification

Un modèle de modification est un ensemble de paramètres de configuration dans Change Manager, qui définissent des éléments tels que les approbations requises, les runbooks disponibles et les options de notification relatives aux demandes de modifications.

> **Note**
>
> AWS fournit un exemple de modèle de modification Hello World, que vous pouvez utiliser pour tester Change Manager, une fonctionnalité de AWS Systems Manager. Toutefois, vous créez vos propres modèles de modifications pour définir les modifications que vous voulez autoriser sur les ressources de votre organisation ou de votre compte.

Les modifications apportées lors de l'exécution d'un flux de travail de runbook sont basées sur le contenu d'un runbook Automation. Dans chaque modèle de modification créé, vous pouvez inclure un ou plusieurs runbooks Automation que l'utilisateur qui effectue une demande de modification peut choisir d'exécuter pendant la mise à jour. Vous pouvez également créer des modèles de modifications qui autorisent les demandeurs à choisir n'importe quel runbook Automation disponible pour la demande de modification.

Pour créer un modèle de modification, vous pouvez utiliser l'option Builder (Générateur) sur la page de la console Create template (Créer un modèle). En variante, vous pouvez utiliser l'option Editor (Éditeur) pour créer manuellement du contenu JSON ou YAML avec la configuration voulue pour votre flux de travail de runbook. Vous pouvez aussi utiliser un outil de ligne de commande pour créer un modèle de modification, avec le contenu JSON pour le modèle de modification stocké dans un fichier externe.

Rubriques

Essayez le modèle de AWS gestion des **`Hello World`** modifications

Vous pouvez utiliser l'exemple de modèle de modification`AWS-HelloWorldChangeTemplate`, qui utilise l'exemple de manuel d'automatisation`AWS-HelloWorld`, pour tester le processus de révision et d'approbation une fois la configuration terminéeChange Manager, une fonctionnalité de AWS Systems Manager. Ce modèle est conçu pour tester ou vérifier vos autorisations configurées, les affectations d'approbateur et le processus d'approbation. L'approbation d'utiliser ce modèle de modification dans votre organisation ou votre compte a déjà été fournie par AWS. Toutefois, toute demande de modification basée sur ce modèle de modification doit toujours être approuvée par les vérificateurs de votre organisation ou de votre compte.

Le résultat du flux de travail de runbook associé à ce modèle ne consiste pas à apporter des modifications à une ressource, mais à imprimer un message dans la sortie d'une étape d'automatisation.

Avant de commencer

Avant de commencer, assurez-vous que vous avez terminé les tâches suivantes :

- Si vous utilisez AWS Organizations pour gérer le changement au sein d'une organisation, effectuez les tâches de configuration de l'organisation décrites dansConfiguration de Change Manager pour une organisation (compte de gestion).
- Configurez Change Manager pour votre compte d'administrateur délégué ou votre compte unique, de la façon décrite dans Configuration d'options et de bonnes pratiques Change Manager.

> **Note**
>
> Si vous avez activé l'option de bonne pratique Exiger des surveillances pour tous les modèles dans vos paramètres Change Manager, désactivez-la momentanément pendant que vous testez le modèle de modification Hello World.

Pour essayer le modèle de modification AWS géré par Hello World

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Change Manager.
3. Sélectionnez Créer une demande.

4. Sélectionnez le modèle de modification nommé `AWS-HelloWorldChangeTemplate`, puis Suivant.
5. Pour Nom de la demande de modification, saisissez un nom permettant d'identifier sa fonction, **`MyChangeRequestTest`** par exemple.
6. Pour les autres étapes de création de votre demande de modification, veuillez consulter Création de demandes de modifications.

Étapes suivantes

Pour obtenir des informations sur l'approbation de demandes de modifications, veuillez consulter Vérifier et approuver ou rejeter les demandes de modifications.

Pour afficher le statut et les résultats de votre demande de modification, sélectionnez le nom de votre demande de modification sous l'onglet Requests (Demandes) dans Change Manager.

Création de modèles de modification

Un modèle de modification est un ensemble de paramètres de configuration dans Change Manager, qui définissent des éléments tels que les approbations requises, les runbooks disponibles et les options de notification relatives aux demandes de modifications.

Vous pouvez créer des modèles de modifications pour vos opérations dans Change Manager, une fonctionnalité de AWS Systems Manager à l'aide de la console, qui inclut les options générateur et éditeur, ou des outils de ligne de commande.

Rubriques

- À propos des approbations dans vos modèles de modification
- Création de modèles de modifications à l'aide du générateur
- Création de modèles de modifications à l'aide de l'éditeur
- Création de modèles de modifications à l'aide des outils de ligne de commande

À propos des approbations dans vos modèles de modification

Pour chaque modèle de modification que vous créez, vous pouvez spécifier jusqu'à cinq niveaux d'approbation pour les demandes de modification créées à partir de ce modèle. Pour chacun de ces niveaux, vous pouvez désigner jusqu'à cinq approbateurs potentiels. Un approbateur n'est pas limité à un seul utilisateur. Vous pouvez également spécifier un groupe IAM ou un rôle IAM comme approbateur individuel. Pour les groupes IAM et les rôles IAM, un ou plusieurs utilisateurs

appartenant au groupe ou au rôle peuvent fournir des approbations afin de recevoir le nombre total d'approbations requises pour une demande de modification. Vous pouvez également spécifier un nombre d'approbateurs supérieur à celui requis par votre modèle de modification.

Change Manager prend en charge deux approches principales en matière d'approbations : les approbations par niveau et les approbations par ligne. Une combinaison des deux types est également possible dans certaines situations. Nous vous recommandons de n'utiliser que des approbations par niveau dans vos opérations Change Manager.

Per-level approvals

Recommandé À compter du 23 janvier 2023, Change Manager prend en charge les approbations par niveau. Dans ce modèle, pour chaque niveau d'approbation de votre modèle de modifications, vous spécifiez d'abord le nombre d'approbations requises pour ce niveau. Ensuite, vous spécifiez au moins autant d'approbateurs pour le niveau et pouvez en spécifier d'autres. Toutefois, seul le nombre d'approbateurs par niveau que vous spécifiez doit approuver la demande de modifications. Par exemple, vous pouvez spécifier cinq approbateurs, mais demander trois approbations.

Pour des exemples avec la console et JSON de ce type d'approbation, consultez the section called “Exemple de configuration d'approbation par niveau”.

Per-line approvals

Pris en charge pour des raisons de rétrocompatibilité. La version initiale des approbations par ligne uniquement prises en charge par Change Manager. Dans ce modèle, chaque approbateur spécifié pour un niveau d'approbation est représenté par une ligne d'approbation. Chaque approbateur devait approuver une demande de modification pour qu'elle soit approuvée à ce niveau. Avant le 23 janvier 2023, il s'agissait du seul modèle pris en charge pour les approbations. Les modèles de modification créés avant cette date continuent de prendre en charge les approbations par ligne, mais nous vous recommandons d'utiliser plutôt des approbations par niveau.

Pour des exemples avec la console et JSON de ce type d'approbation, consultez the section called “Exemple de configuration d'approbation par ligne”.

Combined per-line and per-level approvals

Non recommandé. Dans la console, l'onglet Générateur ne prend plus en charge l'ajout des approbations par ligne. Toutefois, dans certains cas, vous pouvez obtenir des approbations à la fois par ligne et par niveau dans un modèle de modification. Cela peut se produire si vous mettez

à jour un modèle de modification créé avant le 23 janvier 2023, ou si vous créez ou mettez à jour un modèle de modification en modifiant manuellement son contenu YAML,

Pour des exemples avec la console et JSON de ce type d'approbation, consultez the section called “Exemple de configuration d'approbation combinée par niveau et par ligne”.

> **Important**
>
> Bien qu'il soit possible de créer un modèle de modification combinant des approbations par ligne et par niveau, cette configuration n'est ni recommandée ni nécessaire. Le type d'approbation nécessitant le plus d'approbations (approbations par ligne ou par niveau) est prioritaire. Par exemple :
>
> - Si un modèle de modification spécifie trois approbations par niveau mais cinq approbations par ligne, cinq approbations sont requises.
> - Si un modèle de modification spécifie quatre approbations par niveau mais deux approbations par ligne, quatre approbations sont requises.

Vous pouvez créer un niveau qui inclut des approbations par ligne et par niveau en modifiant le contenu YAML ou JSON manuellement. L'onglet Générateur affiche ensuite des commandes permettant de spécifier le nombre d'approbations requis à la fois pour le niveau et pour les lignes individuelles. Toutefois, les nouveaux niveaux que vous ajoutez à l'aide de la console ne prennent toujours en charge que les configurations d'approbation par niveau.

Notifications et refus de demandes de modifications

Notifications Amazon SNS

Lorsqu'une demande de modifications est créée, des notifications sont envoyées aux abonnés de la rubrique Amazon Simple Notification Service (Amazon SNS) qui a été désignée pour les notifications d'approbation à ce niveau.. Vous pouvez spécifier la rubrique de notification dans le modèle de modification ou autoriser l'utilisateur qui crée la demande de modifications à en spécifier une.

Une fois que le nombre minimum d'approbations requises est reçu à un niveau, des notifications sont envoyées aux approbateurs abonnés à la rubrique Amazon SNS pour le niveau suivant, et ainsi de suite.

> **Important**
>
> Assurez-vous que les rôles, les groupes et les utilisateurs IAM que vous désignez ensemble fournissent suffisamment d'approbateurs pour atteindre le nombre d'approbations requis que vous spécifiez. Par exemple, si vous ne désignez qu'un seul groupe IAM comme approbateur contenant trois utilisateurs, vous ne pouvez pas spécifier que cinq approbations sont obligatoires à ce niveau, mais seulement trois ou moins.

Refus de demandes de modification

Quel que soit le nombre de niveaux d'approbation et d'approbateurs que vous spécifiez, un seul refus d'une demande de modification est requis pour empêcher le flux de travail du runbook pour cette demande de se produire.

Exemples de types d'approbation dans Change Manager

Les exemples suivants illustrent la vue de la console et le contenu JSON pour les trois types de types d'approbation dans Change Manager.

Rubriques

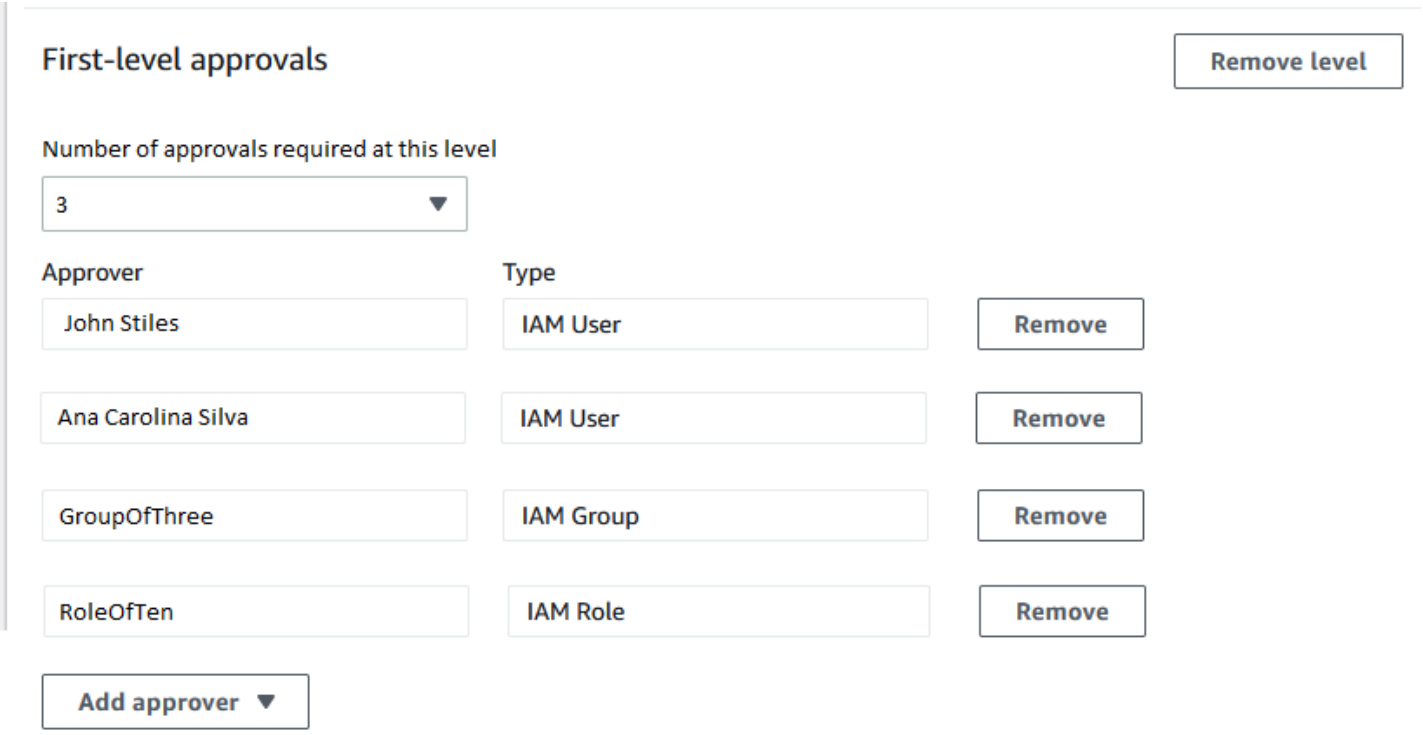
- Exemple de configuration d'approbation par niveau
- Exemple de configuration d'approbation par ligne
- Exemple de configuration d'approbation combinée par niveau et par ligne

Exemple de configuration d'approbation par niveau

Dans la configuration du niveau d'approbation par niveau présentée dans l'image suivante, trois approbations sont requises. Ces approbations peuvent provenir de n'importe quelle combinaison d'utilisateurs, de groupes et de rôles IAM spécifiés en tant qu'approbateurs. Les approbateurs spécifiés incluent deux utilisateurs IAM (John Stiles et Ana Carolina Silva), un groupe d'utilisateurs composé de trois membres (`GroupOfThree`) et un rôle utilisateur représentant dix utilisateurs (`RoleOfTen`).

Si les trois utilisateurs du groupe `GroupOfThree` approuvent la demande de modification, celle-ci est approuvée pour ce niveau. Il n'est pas nécessaire de recevoir l'approbation de chaque utilisateur, groupe ou rôle. Le nombre minimum d'approbations peut provenir de n'importe quelle combinaison

d'approbateurs spécifiés. Nous recommandons des approbations par niveau pour vos opérations dans Change Manager.



L'exemple suivant illustre une partie du code YAML pour cette configuration.

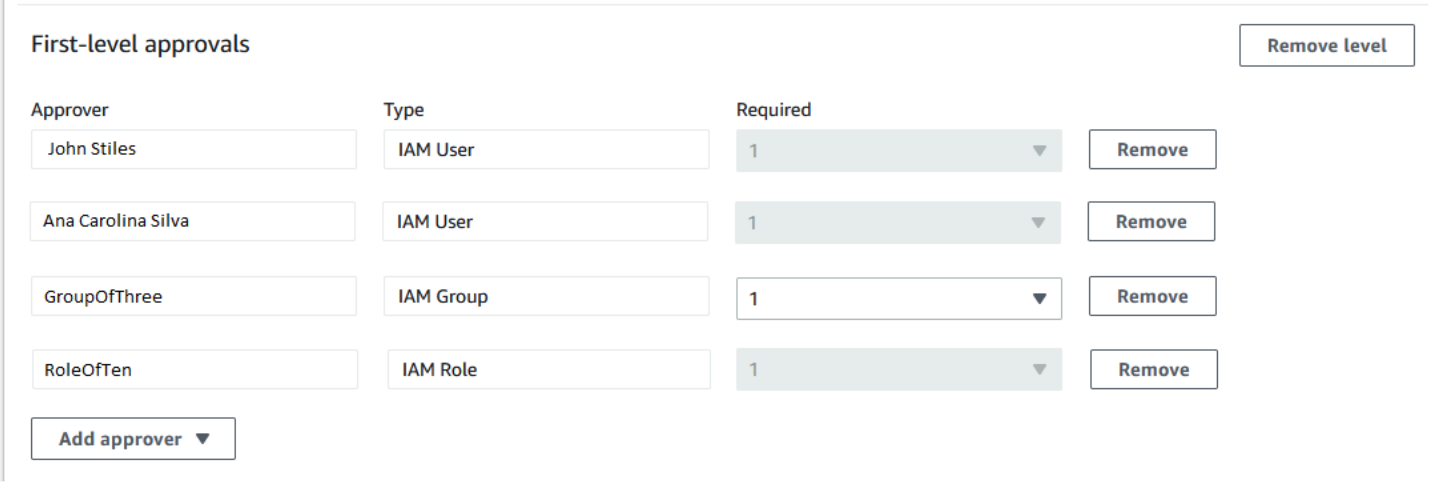
> **Note**
>
> Cette version du code YAML inclut une entrée supplémentaire, `MinRequiredApprovals` (commençant par un `M` majuscule). La valeur de cette entrée indique le nombre d'approbations requises parmi tous les réviseurs disponibles. Notez également que la valeur `minRequiredApprovals` (commençant par un `m` minuscule) de chaque approbateur de la liste `Approvers` est de `0` (zéro). Cela indique que l'approbateur peut contribuer aux approbations globales, mais qu'il n'est pas obligé de le faire.

```
schemaVersion: "0.3"
emergencyChange: false
autoApprovable: false
mainSteps:
  - name: ApproveAction1
    action: aws:approve
```

```
    timeoutSeconds: 604800
    inputs:
      Message: Please approve this change request
      MinRequiredApprovals: 3
      EnhancedApprovals:
        Approvers:
          - approver: John Stiles
            type: IamUser
            minRequiredApprovals: 0
          - approver: Ana Carolina Silva
            type: IamUser
            minRequiredApprovals: 0
          - approver: GroupOfThree
            type: IamGroup
            minRequiredApprovals: 0
          - approver: RoleOfTen
            type: IamRole
            minRequiredApprovals: 0
templateInformation: >
  #### What is the purpose of this change?
    //truncated
```

Exemple de configuration d'approbation par ligne

Dans la configuration du niveau d'approbation présentée dans l'image suivante, quatre approbateurs sont spécifiés. Il s'agit notamment de deux utilisateurs IAM (John Stiles et Ana Carolina Silva), d'un groupe d'utilisateurs composé de trois membres (`GroupOfThree`) et d'un rôle utilisateur représentant dix utilisateurs (`RoleOfTen`). Les approbations par ligne sont prises en charge pour des raisons de rétrocompatibilité, mais elles ne sont pas recommandées.



Pour que la demande de modifications soit approuvée dans cette configuration d'approbation par ligne, elle doit être approuvée par toutes les lignes d'approbation : John Stiles, Ana Carolina Silva, un membre du groupe `GroupOfThree` et un membre du rôle `RoleOfTen`.

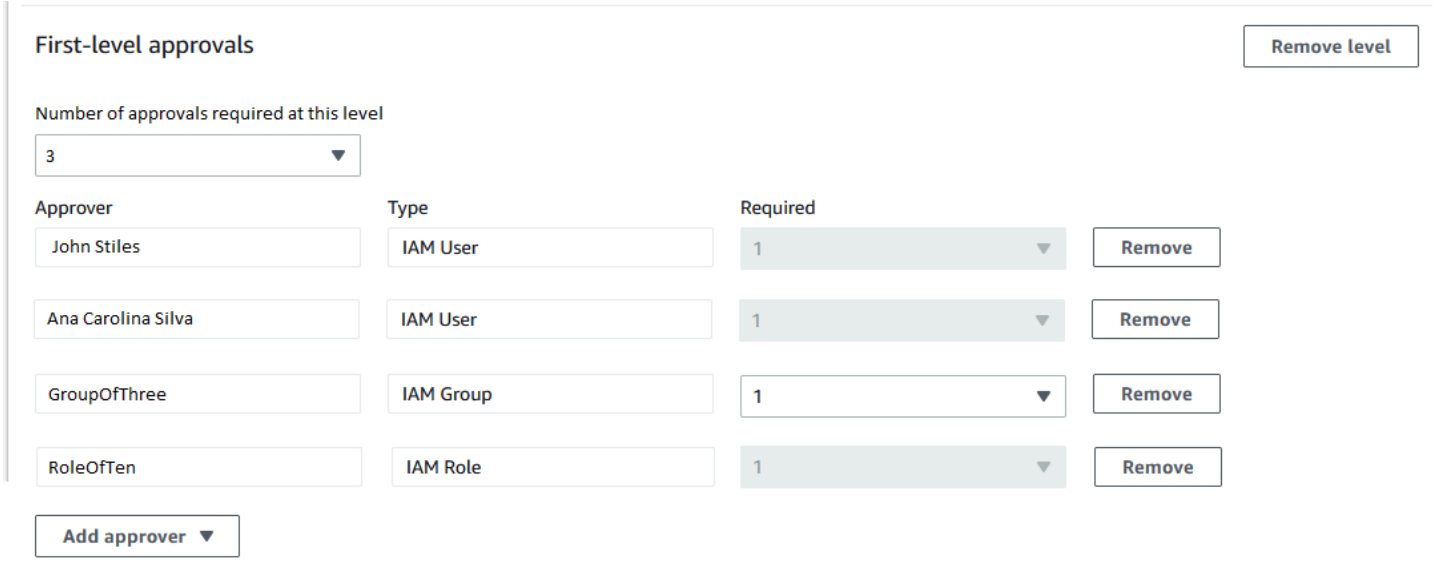
L'exemple suivant illustre une partie du code YAML pour cette configuration.

> **Note**
>
> Notez que la valeur de chaque approbateur `minRequiredApprovals` est de 1. Cela indique qu'une approbation est requise de la part de chaque approbateur.

```
schemaVersion: "0.3"
emergencyChange: false
autoApprovable: false
mainSteps:
  - name: ApproveAction1
    action: aws:approve
    timeoutSeconds: 10000
    inputs:
      Message: Please approve this change request
      EnhancedApprovals:
        Approvers:
          - approver: John Stiles
            type: IamUser
            minRequiredApprovals: 1
          - approver: Ana Carolina Silva
            type: IamUser
            minRequiredApprovals: 1
          - approver: GroupOfThree
            type: IamGroup
            minRequiredApprovals: 1
          - approver: RoleOfTen
            type: IamRole
            minRequiredApprovals: 1
executableRunBooks:
  - name: AWS-HelloWorld
    version: $DEFAULT
templateInformation: >
  #### What is the purpose of this change?
    //truncated
```

Exemple de configuration d'approbation combinée par niveau et par ligne

Dans la configuration d'approbation combinée par niveau et par ligne présentée dans l'image suivante, trois approbations sont spécifiées pour le niveau, mais quatre approbations sont spécifiées pour les approbations par poste. Quel que soit le type d'approbation qui nécessite le plus d'approbations, cette configuration nécessite quatre approbations. L'approbation combinée par niveau et par ligne n'est pas recommandée.




```
schemaVersion: "0.3"
emergencyChange: false
autoApprovable: false
mainSteps:
  - name: ApproveAction1
    action: aws:approve
    timeoutSeconds: 604800
    inputs:
      Message: Please approve this change request
      MinRequiredApprovals: 3
      EnhancedApprovals:
        Approvers:
          - approver: John Stiles
            type: IamUser
            minRequiredApprovals: 1
          - approver: Ana Carolina Silva
            type: IamUser
            minRequiredApprovals: 1
          - approver: GroupOfThree
```

```
            type: IamGroup
            minRequiredApprovals: 1
          - approver: RoleOfTen
            type: IamRole
            minRequiredApprovals: 1
templateInformation: >
  #### What is the purpose of this change?
    //truncated
```

Rubriques

- Création de modèles de modifications à l'aide du générateur
- Création de modèles de modifications à l'aide de l'éditeur
- Création de modèles de modifications à l'aide des outils de ligne de commande

Création de modèles de modifications à l'aide du générateur

En utilisant le générateur de modèles de modifications dans Change Manager, une fonctionnalité de AWS Systems Manager, vous pouvez configurer le flux de travail de runbook défini dans votre modèle de modification, sans utiliser la syntaxe JSON ou YAML. Après avoir spécifié vos options, le système convertit votre entrée au format YAML, que Systems Manager peut utiliser pour exécuter des flux de travail de runbook.

Pour créer un modèle de modification à l'aide du générateur

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Change Manager.
3. Sélectionnez Create template (Créer un modèle).
4. Pour Name (Nom), saisissez un nom pour le modèle permettant d'identifier sa fonction, **`UpdateEC2LinuxAMI`** par exemple.
5. Dans la section Change template details (Détails du modèle de modification), procédez comme suit :

   - Pour Description, expliquez brièvement comment et quand le modèle de modification que vous créez doit être utilisé.

Cette description permet aux utilisateurs qui créent des demandes de modifications de déterminer s'ils utilisent le bon modèle de modification. Elle aide les personnes qui vérifient les demandes de modifications à comprendre si la demande doit être approuvée.

- Pour Change template type (Type de modèle de modification), spécifiez si vous créez un modèle de modification standard ou un modèle de modification d'urgence.

  Un modèle de modification d'urgence est utilisé dans les situations où une modification doit être apportée même si les modifications sont bloquées par un événement du calendrier utilisé par AWS Systems Manager Change Calendar. Les demandes de modifications créées à partir d'un modèle de modification d'urgence doivent toujours être approuvées par ses approbateurs désignés, mais les modifications demandées peuvent quand même s'exécuter même lorsque le calendrier est bloqué.

- Pour Runbook options (Options de runbook), spécifiez les runbooks que les utilisateurs peuvent choisir lorsqu'ils créent une demande de modification. Vous pouvez ajouter un seul ou plusieurs runbooks. En variante, vous pouvez autoriser les demandeurs à spécifier le runbook à utiliser. Dans tous ces cas, la demande de modification ne peut contenir qu'un seul runbook.

- Pour Runbook, sélectionnez les noms et les versions de runbooks que les utilisateurs peuvent choisir pour leurs demandes de modifications. Indépendamment du nombre de runbooks que vous ajoutez au modèle de modification, un seul d'entre eux peut être sélectionné par demande de modification.

  Vous n'avez pas à spécifier de runbook si vous avez précédemment choisi Any runbook can be used (N'importe quel runbook peut être utilisé).

  > **Tip**
  >
  > Sélectionnez un runbook et sa version, puis sélectionnez View (Afficher) pour examiner le contenu du runbook dans l'interface Systems Manager Documents.

6. Dans la section Template information (Informations sur le modèle), sélectionnez Markdown pour saisir des informations pour les utilisateurs qui créent des demandes de modifications à partir de ce modèle de modification. Un ensemble de questions vous est fourni, que vous pouvez inclure pour les utilisateurs qui créent des demandes de modifications. Vous pouvez aussi ajouter d'autres informations et questions à la place.

> **Note**
>
> Markdown est 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.

Nous vous recommandons de répondre aux questions que les utilisateurs se posent à propos de leurs demandes de modifications afin d'aider les approbateurs à décider s'ils acceptent ou non chaque demande de modification, notamment la liste des étapes manuelles à exécuter dans le cadre de la modification et un plan de restauration.

> **Tip**
>
> Basculez entre Hide preview (Masquer l'aperçu) et Show preview (Afficher l'aperçu) pour voir à quoi ressemble votre contenu lorsque vous le créez.

7. Dans la section Change request approvals (Approbations de demande de modification), procédez de la manière suivante :

   - (Facultatif) Si vous voulez autoriser l'exécution automatique des demandes de modifications créées à partir de ce modèle de modification, sans vérification par des approbateurs (à l'exception des événements de gel des modifications), sélectionnez Enable auto-approval (Activer l'approbation automatique).

     > **Note**
     >
     > L'activation des approbations automatiques dans un modèle de modification fournit aux utilisateurs l'option de contourner les vérificateurs. Ils peuvent toujours choisir de spécifier des vérificateurs lors de la création d'une demande de modification. Vous devez donc toujours spécifier des options de vérificateurs dans le modèle de modification.

> **Important**
>
> Si vous activez l'approbation automatique d'un modèle de modification, les utilisateurs peuvent utiliser ce modèle pour envoyer des demandes de modifications n'exigeant pas de vérification par des vérificateurs avant leur exécution (à l'exception des approbateurs d'événements de gel des modifications). Si vous voulez empêcher un utilisateur, un groupe ou un rôle IAM particulier d'envoyer des demandes d'approbation automatique, vous pouvez utiliser une condition dans une politique IAM. Pour plus d'informations, consultez Contrôler l'accès aux flux de travail de runbook d'approbation automatique.

- Dans Nombre d'approbations requises à ce niveau, choisissez le nombre d'approbations que les demandes de modifications créées à partir de ce modèle de modification doivent recevoir pour ce niveau.
- Pour ajouter des approbateurs de premier niveau obligatoires, sélectionnez Add approver (Ajouter un approbateur), puis sélectionnez l'une des options suivantes :
  - Approbateurs spécifiés par le modèle : sélectionnez un ou plusieurs utilisateurs, groupes ou rôles AWS Identity and Access Management (IAM) de votre compte pour approuver les demandes de modifications créées à partir de ce modèle de modification. Toutes les demandes de modifications créées avec ce modèle doivent être vérifiées et approuvées par chaque approbateur spécifié.
  - Approbateurs spécifiés par la demande : l'utilisateur qui effectue la demande de modification spécifie les vérificateurs au moment de la demande, et il peut choisir parmi une liste d'utilisateurs dans votre compte.

    Le nombre que vous saisissez dans la colonne Required (Obligatoire) détermine le nombre de vérificateurs à spécifier pour une demande de modification utilisant ce modèle de modification.

> **Important**
>
> Avant le 23 janvier 2023, l'onglet Générateur permettait de spécifier uniquement les approbations par ligne. Les nouveaux modèles de modification et les nouveaux niveaux que vous ajoutez à des modèles de modification existants à l'aide de l'onglet Générateur ne prennent en charge que les approbations par niveau. Nous vous

recommandons de n'utiliser que des approbations par niveau dans vos opérations dans Change Manager.
Pour plus d'informations, consultez À propos des approbations dans vos modèles de modification.

- Pour SNS topic to notify approvers (Rubrique SNS utilisée pour informer les approbateurs), procédez comme suit :

  1. Sélectionnez l'une des options suivantes pour spécifier la rubrique Amazon Simple Notification Service (Amazon SNS) de votre compte à utiliser pour envoyer aux approbateurs des notifications indiquant qu'une demande de modification est prête à être vérifiée :

     - Saisir un Amazon Resource Name (ARN) SNS : pour ARN de rubrique, saisissez l'ARN d'une rubrique Amazon SNS existante. Cette rubrique peut se trouver dans n'importe quel compte de votre organisation.
     - Sélectionner une rubrique SNS existante : pour Rubrique de notification cible, sélectionnez l'ARN d'une rubrique Amazon SNS existante dans votre Compte AWS actuel. (Cette option n'est pas disponible si vous n'avez pas encore créé de rubrique Amazon SNS dans votre Compte AWS et Région AWS.)
     - Spécifier la rubrique SNS lors de la création de la demande de modification : l'utilisateur qui crée une demande de modification peut spécifier la rubrique Amazon SNS à utiliser pour les notifications.

     > **Note**
     >
     > La rubrique Amazon SNS que vous sélectionnez doit être configurée pour spécifier les notifications qu'elle envoie, ainsi que les abonnés auxquels elles sont envoyées. Sa politique d'accès doit également octroyer des autorisations à Systems Manager de sorte que Change Manager puisse envoyer des notifications. Pour plus d'informations, consultez Configuration des rubriques Amazon SNS pour les notifications Change Manager.

  2. Sélectionnez Ajouter une notification.

8. (Facultatif) Pour ajouter un niveau supplémentaire d'approbateurs, sélectionnez Add approval level (Ajouter un niveau d'approbation) et sélectionnez entre les approbateurs spécifiés par le modèle et les approbateurs spécifiés par la demande pour ce niveau. Ensuite, sélectionnez une rubrique SNS à utiliser pour informer ce niveau d'approbateurs.

Une fois toutes les approbations reçues par les approbateurs de premier niveau, c'est au tour des approbateurs de deuxième niveau d'être informés, etc.

Vous pouvez ajouter cinq niveaux d'approbateurs maximum dans chaque modèle. Par exemple, vous pouvez exiger des approbations des utilisateurs occupant des rôles techniques pour le premier niveau, puis des approbations managériales pour le deuxième niveau.

9. Dans la section Surveillance, pour qu'une CloudWatch alarme soit surveillée, entrez le nom d'une CloudWatch alarme Amazon dans le compte courant afin de suivre la progression des flux de travail Runbook basés sur ce modèle.

> **Tip**
>
> Pour créer une nouvelle alarme ou pour revoir les paramètres d'une alarme que vous souhaitez spécifier, choisissez Ouvrir la CloudWatch console Amazon. Pour plus d'informations sur l'utilisation des CloudWatch alarmes, consultez la section Utilisation des CloudWatch alarmes dans le guide de CloudWatch l'utilisateur Amazon.

10. Dans la section Notifications, procédez comme suit :

    1. Sélectionnez l'une des options suivantes pour spécifier la rubrique Amazon SNS de votre compte à utiliser pour envoyer des notifications à propos des demandes de modifications créées à l'aide de ce modèle de modification :

       - Saisir un Amazon Resource Name (ARN) SNS : pour ARN de rubrique, saisissez l'ARN d'une rubrique Amazon SNS existante. Cette rubrique peut se trouver dans n'importe quel compte de votre organisation.
       - Sélectionner une rubrique SNS existante : pour Rubrique de notification cible, sélectionnez l'ARN d'une rubrique Amazon SNS existante dans votre Compte AWS actuel. (Cette option n'est pas disponible si vous n'avez pas encore créé de rubrique Amazon SNS dans votre Compte AWS et Région AWS.)

       > **Note**
       >
       > La rubrique Amazon SNS que vous sélectionnez doit être configurée pour spécifier les notifications qu'elle envoie, ainsi que les abonnés auxquels elles sont envoyées. Sa politique d'accès doit également octroyer des autorisations à Systems Manager de sorte que Change Manager puisse envoyer des notifications. Pour plus d'informations,

consultez Configuration des rubriques Amazon SNS pour les notifications Change Manager.

2. Sélectionnez Ajouter une notification.

11. (Facultatif) Dans la section Tags (Balises), appliquez une ou plusieurs paires nom/valeur de clé de balise au modèle de modification.

    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 étiqueter un modèle de modification afin d'identifier le type de modification qu'il effectue et l'environnement dans lequel il s'exécute. Dans ce cas, vous pouvez spécifier les paires nom/valeur de clé suivantes :

    - `Key=TaskType,Value=InstanceRepair`
    - `Key=Environment,Value=Production`

    Pour plus d'informations sur le balisage des ressources Systems Manager, consultez Balisage des ressources Systems Manager.

12. Sélectionnez Save and preview (Enregistrer et prévisualiser).

13. Vérifiez les détails du modèle de modification que vous créez.

    Si vous voulez apporter des modifications au modèle de modification avant de l'envoyer en vérification, sélectionnez Actions, Edit (Actions, modifier).

    Si le contenu du modèle de modification vous convient, sélectionnez Submit for review (Envoyer en vérification). Les utilisateurs de votre organisation ou de votre compte qui ont été spécifiés comme vérificateurs de modèles sous l'onglet Settings (Paramètres) dans Change Manager sont informés qu'un nouveau modèle de modification est en attente de vérification.

    Si une rubrique Amazon SNS a été spécifiée pour les modèles de modifications, des notifications sont envoyées lorsque le modèle de modification est rejeté ou approuvé. Si vous ne recevez pas de notifications liées à ce modèle de modification, vous pouvez revenir à Change Manager plus tard afin de vérifier son statut.

Création de modèles de modifications à l'aide de l'éditeur

Suivez les étapes décrites dans cette rubrique pour configurer un modèle de modification dans une fonctionnalité de Change Manager AWS Systems Manager, en saisissant du code JSON ou YAML au lieu d'utiliser les commandes de console.

Pour créer un modèle de modification à l'aide d'Editor

1. Dans le panneau de navigation, sélectionnez Change Manager.
2. Sélectionnez Create template (Créer un modèle).
3. Pour Name (Nom), saisissez un nom pour le modèle permettant d'identifier sa fonction, **`RestartEC2LinuxInstance`** par exemple.
4. Au-dessus de Change template details (Détails du modèle de modification), sélectionnez Editor (Éditeur).
5. Dans la section Document editor (Éditeur de document), sélectionnez Edit (Modifier), puis saisissez le contenu JSON ou YAML pour votre modèle de modification.

   Voici un exemple.

   > **Note**
   >
   > Le paramètre `minRequiredApprovals` permet de spécifier combien de réviseurs à un niveau spécifié doivent approuver une demande de modification créée à l'aide de ce modèle.
   > Cet exemple illustre deux niveaux d'approbation. Vous pouvez spécifier jusqu'à cinq niveaux d'approbation, mais un seul niveau est requis.
   > Au premier niveau, l'utilisateur spécifique « John-Doe » doit approuver chaque demande de modification. Après cela, trois membres du rôle de l'IAM `Admin` doivent approuver la demande de modification.
   > Pour plus d'informations sur les modèles de modification, consultez À propos des approbations dans vos modèles de modification.

   YAML

```
description: >-
  This change template demonstrates the feature set available for creating
  change templates for Change Manager. This template starts a Runbook workflow
```

```
  for the Automation runbook called AWS-HelloWorld.
templateInformation: >
  ### Document Name: HelloWorldChangeTemplate

  ## What does this document do?

  This change template demonstrates the feature set available for creating
  change templates for Change Manager. This template starts a Runbook workflow
  for the Automation runbook called AWS-HelloWorld.

  ## Input Parameters

  * ApproverSnsTopicArn: (Required) Amazon Simple Notification Service ARN for
  approvers.

  * Approver: (Required) The name of the approver to send this request to.

  * ApproverType: (Required) The type of reviewer.
    * Allowed Values: IamUser, IamGroup, IamRole, SSOGroup, SSOUser

  ## Output Parameters

  This document has no outputs
schemaVersion: '0.3'
parameters:
  ApproverSnsTopicArn:
    type: String
    description: Amazon Simple Notification Service ARN for approvers.
  Approver:
    type: String
    description: IAM approver
  ApproverType:
    type: String
    description: >-
      Approver types for the request. Allowed values include IamUser, IamGroup,
      IamRole, SSOGroup, and SSOUser.
executableRunBooks:
  - name: AWS-HelloWorld
    version: '1'
emergencyChange: false
autoApprovable: false
mainSteps:
  - name: ApproveAction1
    action: 'aws:approve'
```

```
    timeoutSeconds: 3600
    inputs:
      Message: >-
        A sample change request has been submitted for your review in Change
        Manager. You can approve or reject this request.
      EnhancedApprovals:
        NotificationArn: '{{ ApproverSnsTopicArn }}'
        Approvers:
          - approver: John-Doe
            type: IamUser
            minRequiredApprovals: 1
  - name: ApproveAction2
    action: 'aws:approve'
    timeoutSeconds: 3600
    inputs:
      Message: >-
        A sample change request has been submitted for your review in Change
        Manager. You can approve or reject this request.
      EnhancedApprovals:
        NotificationArn: '{{ ApproverSnsTopicArn }}'
        Approvers:
          - approver: Admin
            type: IamRole
            minRequiredApprovals: 3
```

JSON

```
{
   "description": "This change template demonstrates the feature set available
 for creating
  change templates for Change Manager. This template starts a Runbook workflow
  for the Automation runbook called AWS-HelloWorld",
   "templateInformation": "### Document Name: HelloWorldChangeTemplate\n\n
    ## What does this document do?\n
    This change template demonstrates the feature set available for creating
 change templates for Change Manager.
    This template starts a Runbook workflow for the Automation runbook called
 AWS-HelloWorld.\n\n
    ## Input Parameters\n* ApproverSnsTopicArn: (Required) Amazon Simple
 Notification Service ARN for approvers.\n
    * Approver: (Required) The name of the approver to send this request to.\n
    * ApproverType: (Required) The type of reviewer.  * Allowed Values: IamUser,
 IamGroup, IamRole, SSOGroup, SSOUser\n\n
```

```
    ## Output Parameters\nThis document has no outputs\n",
  "schemaVersion": "0.3",
  "parameters": {
     "ApproverSnsTopicArn": {
        "type": "String",
        "description": "Amazon Simple Notification Service ARN for approvers."
     },
     "Approver": {
        "type": "String",
        "description": "IAM approver"
     },
     "ApproverType": {
        "type": "String",
        "description": "Approver types for the request. Allowed values include
IamUser, IamGroup, IamRole, SSOGroup, and SSOUser."
     }
  },
  "executableRunBooks": [
     {
        "name": "AWS-HelloWorld",
        "version": "1"
     }
  ],
  "emergencyChange": false,
  "autoApprovable": false,
  "mainSteps": [
     {
        "name": "ApproveAction1",
        "action": "aws:approve",
        "timeoutSeconds": 3600,
        "inputs": {
           "Message": "A sample change request has been submitted for your
review in Change Manager. You can approve or reject this request.",
           "EnhancedApprovals": {
              "NotificationArn": "{{ ApproverSnsTopicArn }}",
              "Approvers": [
                 {
                    "approver": "John-Doe",
                    "type": "IamUser",
                    "minRequiredApprovals": 1
                 }
              ]
           }
        }
```

```
      },
        {
         "name": "ApproveAction2",
         "action": "aws:approve",
         "timeoutSeconds": 3600,
         "inputs": {
            "Message": "A sample change request has been submitted for your
 review in Change Manager. You can approve or reject this request.",
            "EnhancedApprovals": {
               "NotificationArn": "{{ ApproverSnsTopicArn }}",
               "Approvers": [
                  {
                     "approver": "Admin",
                     "type": "IamRole",
                     "minRequiredApprovals": 3
                  }
               ]
            }
         }
      }
   ]
}
```

6. Sélectionnez Save and preview (Enregistrer et prévisualiser).
7. Vérifiez les détails du modèle de modification que vous créez.

   Si vous voulez apporter des modifications au modèle de modification avant de l'envoyer en vérification, sélectionnez Actions, Edit (Actions, modifier).

   Si le contenu du modèle de modification vous convient, sélectionnez Submit for review (Envoyer en vérification). Les utilisateurs de votre organisation ou de votre compte qui ont été spécifiés comme vérificateurs de modèles sous l'onglet Settings (Paramètres) dans Change Manager sont informés qu'un nouveau modèle de modification est en attente de vérification.

   Si une rubrique Amazon Simple Notification Service (Amazon SNS) a été spécifiée pour les modèles de modifications, des notifications sont envoyées lorsque le modèle de modification est rejeté ou approuvé. Si vous ne recevez pas de notifications liées à ce modèle de modification, vous pouvez revenir à Change Manager plus tard afin de vérifier son statut.

Création de modèles de modifications à l'aide des outils de ligne de commande

Les procédures suivantes décrivent comment utiliser le AWS Command Line Interface (AWS CLI) (sous Linux ou Windows) ou comment AWS Tools for Windows PowerShell créer une demande de modification dans Change Manager une fonctionnalité de AWS Systems Manager. macOS

Pour créer un modèle de modification

1. Installez et configurez le AWS CLI ou le AWS Tools for PowerShell, si ce n'est pas déjà fait.

   Pour plus d'informations, consultez la section Installation ou mise à jour de la version la plus récente de l' AWS CLI et Installation d' AWS Tools for PowerShell.

2. Créez un fichier JSON sur votre ordinateur local avec un nom tel que `MyChangeTemplate.json`, puis collez le contenu dans votre fichier de modification.

   > **Note**
   >
   > Les modèles de modifications utilisent une version de schéma 0.3 qui n'inclut pas l'intégralité de la prise en charge appliquée aux runbooks Automation.

   Voici un exemple.

   > **Note**
   >
   > Le paramètre `minRequiredApprovals` permet de spécifier combien de réviseurs à un niveau spécifié doivent approuver une demande de modification créée à l'aide de ce modèle.
   > Cet exemple illustre deux niveaux d'approbation. Vous pouvez spécifier jusqu'à cinq niveaux d'approbation, mais un seul niveau est requis.
   > Au premier niveau, l'utilisateur spécifique « John-Doe » doit approuver chaque demande de modification. Après cela, trois membres du rôle de l'IAM `Admin` doivent approuver la demande de modification.
   > Pour plus d'informations sur les modèles de modification, consultez À propos des approbations dans vos modèles de modification.

```
{
```

```
   "description": "This change template demonstrates the feature set available for
 creating
  change templates for Change Manager. This template starts a Runbook workflow
  for the Automation runbook called AWS-HelloWorld",
   "templateInformation": "### Document Name: HelloWorldChangeTemplate\n\n
    ## What does this document do?\n
    This change template demonstrates the feature set available for creating change
 templates for Change Manager.
    This template starts a Runbook workflow for the Automation runbook called AWS-
HelloWorld.\n\n
    ## Input Parameters\n* ApproverSnsTopicArn: (Required) Amazon Simple
 Notification Service ARN for approvers.\n
    * Approver: (Required) The name of the approver to send this request to.\n
    * ApproverType: (Required) The type of reviewer.  * Allowed Values: IamUser,
 IamGroup, IamRole, SSOGroup, SSOUser\n\n
    ## Output Parameters\nThis document has no outputs\n",
  "schemaVersion": "0.3",
  "parameters": {
     "ApproverSnsTopicArn": {
        "type": "String",
        "description": "Amazon Simple Notification Service ARN for approvers."
     },
     "Approver": {
        "type": "String",
        "description": "IAM approver"
     },
     "ApproverType": {
        "type": "String",
        "description": "Approver types for the request. Allowed values include
 IamUser, IamGroup, IamRole, SSOGroup, and SSOUser."
     }
  },
  "executableRunBooks": [
     {
        "name": "AWS-HelloWorld",
        "version": "1"
     }
  ],
  "emergencyChange": false,
  "autoApprovable": false,
  "mainSteps": [
     {
        "name": "ApproveAction1",
        "action": "aws:approve",
```

```
         "timeoutSeconds": 3600,
         "inputs": {
            "Message": "A sample change request has been submitted for your review
 in Change Manager. You can approve or reject this request.",
            "EnhancedApprovals": {
               "NotificationArn": "{{ ApproverSnsTopicArn }}",
               "Approvers": [
                  {
                     "approver": "John-Doe",
                     "type": "IamUser",
                     "minRequiredApprovals": 1
                  }
               ]
            }
         }
      },
        {
         "name": "ApproveAction2",
         "action": "aws:approve",
         "timeoutSeconds": 3600,
         "inputs": {
            "Message": "A sample change request has been submitted for your review
 in Change Manager. You can approve or reject this request.",
            "EnhancedApprovals": {
               "NotificationArn": "{{ ApproverSnsTopicArn }}",
               "Approvers": [
                  {
                     "approver": "Admin",
                     "type": "IamRole",
                     "minRequiredApprovals": 3
                  }
               ]
            }
         }
      }
   ]
}
```

3. Exécutez la commande suivante pour créer le modèle de modification.

   Linux & macOS

   ```
   aws ssm create-document \
       --name MyChangeTemplate \
   ```

```
    --document-format JSON \
    --document-type Automation.ChangeTemplate \
    --content file://MyChangeTemplate.json \
    --tags Key=tag-key,Value=tag-value
```

Windows

```
aws ssm create-document ^
    --name MyChangeTemplate ^
    --document-format JSON ^
    --document-type Automation.ChangeTemplate ^
    --content file://MyChangeTemplate.json ^
    --tags Key=tag-key,Value=tag-value
```

PowerShell

```
$json = Get-Content -Path "C:\path\to\file\MyChangeTemplate.json" | Out-String
New-SSMDocument `
    -Content $json `
    -Name "MyChangeTemplate" `
    -DocumentType "Automation.ChangeTemplate" `
    -Tags "Key=tag-key,Value=tag-value"
```

Pour plus d'informations sur les autres options que vous pouvez spécifier, consultez create-document.

Le système retourne des informations telles que les suivantes.

```
{
   "DocumentDescription":{
      "CreatedDate":1.585061751738E9,
      "DefaultVersion":"1",
      "Description":"Use this template to update an EC2 Linux AMI. Requires one
      approver specified in the template and an approver specified in the
 request.",
      "DocumentFormat":"JSON",
      "DocumentType":"Automation",
      "DocumentVersion":"1",
      "Hash":"0d3d879b3ca072e03c12638d0255ebd004d2c65bd318f8354fcde820dEXAMPLE",
      "HashType":"Sha256",
      "LatestVersion":"1",
```

```
      "Name":"MyChangeTemplate",
      "Owner":"123456789012",
      "Parameters":[
         {
            "DefaultValue":"",
            "Description":"Level one approvers",
            "Name":"LevelOneApprovers",
            "Type":"String"
         },
         {
            "DefaultValue":"",
            "Description":"Level one approver type",
            "Name":"LevelOneApproverType",
            "Type":"String"
         },
   "cloudWatchMonitors": {
      "monitors": [
         "my-cloudwatch-alarm"
      ]
   }
      ],
      "PlatformTypes":[
         "Windows",
         "Linux"
      ],
      "SchemaVersion":"0.3",
      "Status":"Creating",
      "Tags":[

      ]
   }
}
```

Les utilisateurs de votre organisation ou de votre compte qui ont été spécifiés comme vérificateurs de modèles sous l'onglet Settings (Paramètres) dans Change Manager sont informés qu'un nouveau modèle de modification est en attente de vérification.

Si une rubrique Amazon Simple Notification Service (Amazon SNS) a été spécifiée pour les modèles de modifications, des notifications sont envoyées lorsque le modèle de modification est rejeté ou approuvé. Si vous ne recevez pas de notifications liées à ce modèle de modification, vous pouvez revenir à Change Manager plus tard afin de vérifier son statut.

Vérification et approbation de modèles de modifications

Si vous êtes désigné comme réviseur pour les modèles de modification dansChange Manager, une fonctionnalité de AWS Systems Manager, vous êtes averti lorsqu'un nouveau modèle de modification, ou une nouvelle version d'un modèle de modification, attend votre révision. Une rubrique Amazon Simple Notification Service (Amazon SNS) envoie les notifications.

> **Note**
>
> Cette fonctionnalité dépend du fait que votre compte a été, ou non, configuré pour utiliser une rubrique Amazon SNS afin d'envoyer des notifications de vérification de modèle de modification. Pour obtenir des informations sur la spécification d'une rubrique de notification de vérificateur de modèle, veuillez consulter Tâche 1 : configuration de vérificateurs de gestion des identités des utilisateurs et de modèles Change Manager.

Pour consulter le modèle de modification, suivez le lien figurant dans votre notification AWS Management Console, connectez-vous au et suivez les étapes de cette procédure.

Pour vérifier et approuver ou rejeter un modèle de modification

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Change Manager.
3. Dans la section Modèles de modifications au bas de l'onglet Présentation, sélectionnez le nombre dans Vérification en attente.
4. Dans la liste Modèles de modifications, recherchez et sélectionnez le nom du modèle de modification à vérifier.
5. Sur la page récapitulative, vérifiez le contenu proposé du modèle de modification et effectuez l'une des opérations suivantes :

   - Pour approuver le modèle de modification, ce qui autorise son utilisation dans les demandes de modification, sélectionnez Approve (Approuver).
   - Pour rejeter le modèle de modification, ce qui empêche son utilisation dans les demandes de modification, sélectionnez Reject (Rejeter).

Suppression de modèles de modification

Cette rubrique explique comment supprimer les modèles que vous avez créés dans la fonctionnalité Change Manager de Systems Manager. Si vous utilisez Change Manager pour une organisation, cette procédure doit être effectuée à partir de votre compte administrateur délégué.

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Change Manager.
3. Sélectionnez l'onglet Templates (Modèles) .
4. Sélectionnez le nom du modèle à supprimer.
5. Sélectionnez Actions, Delete template (Actions, Supprimer le modèle).
6. Dans le champ de confirmation, saisissez **DELETE**, puis sélectionnez Delete (Supprimer).

Utilisation des demandes de modifications

Une demande de modification est une demande d'exécution d'un runbook Automation dans Change Manager, qui met à jour une ou plusieurs ressources dans votre environnement AWS ou sur site. Une demande de modification est créée à l'aide d'un modèle de modification.

Lorsque vous créez une demande de modification dans Change Manager, une fonctionnalité de AWS Systems Manager, un ou plusieurs approbateurs de votre organisation ou de votre compte doivent vérifier et approuver la demande. Sans les approbations requises, le flux de travail de runbook, qui effectue les modifications demandées, n'est pas autorisé à s'exécuter.

Rubriques

- Création de demandes de modifications
- Vérifier et approuver ou rejeter les demandes de modifications

Création de demandes de modifications

Lorsque vous créez une demande de modification dans Change Manager une fonctionnalité de AWS Systems Manager, le modèle de modification que vous sélectionnez effectue généralement les opérations suivantes :

- Désigne les approbateurs de la demande de modification ou spécifie le nombre d'approbations requises

- Spécifie la rubrique Amazon Simple Notification Service (Amazon SNS) à utiliser pour informer les approbateurs de votre demande de modification.
- Spécifie une CloudWatch alarme Amazon pour surveiller le flux de travail du runbook pour la demande de modification
- Identifie les runbooks Automation que vous pouvez choisir pour effectuer la modification demandée

Dans certains cas, un modèle de modification peut être configuré de sorte à spécifier l'utilisation de votre propre runbook Automation et spécifier qui doit vérifier et approuver la demande.

> **Important**
>
> Si vous utilisez Change Manager au sein d'une organisation, nous vous recommandons de toujours apporter les modifications à partir du compte d'administrateur délégué. Bien qu'il soit possible d'apporter des modifications à partir d'autres comptes de l'organisation, celles-ci ne seront pas signalées ou affichées à partir du compte d'administrateur délégué.

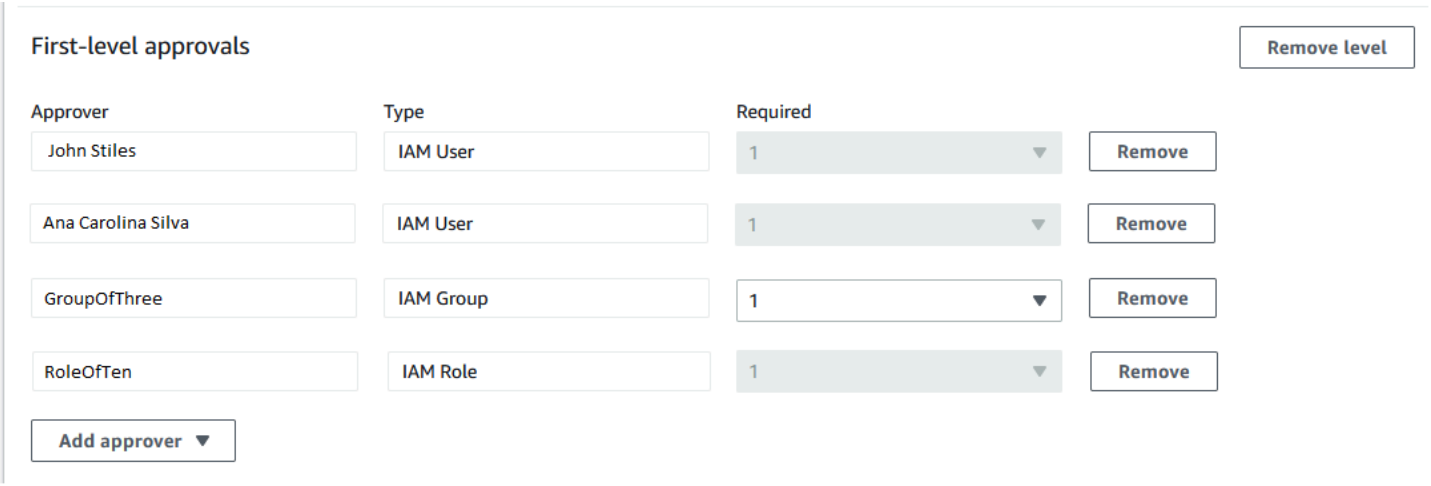
Rubriques

À propos des approbations de demandes de modification

En fonction des exigences spécifiées dans un modèle de modification, les demandes de modification que vous créez à partir de celui-ci peuvent nécessiter des approbations de cinq niveaux au maximum avant que le flux de travail de runbook associé à la demande puisse être exécuté. Pour chacun de ces niveaux, le créateur du modèle pouvait spécifier jusqu'à cinq approbateurs potentiels. Un approbateur n'est pas limité à un seul utilisateur. Dans ce sens, un approbateur peut également être un groupe IAM ou un rôle IAM. Pour les groupes IAM et les rôles IAM, un ou plusieurs utilisateurs appartenant au groupe ou au rôle peuvent fournir des approbations afin de recevoir le nombre total d'approbations requises pour une demande de modification. Les créateurs de modèles peuvent également spécifier un nombre d'approbateurs supérieur à celui requis par le modèle de modification.

Flux de travail d'approbation originaux et mis à jour et/ou approbations

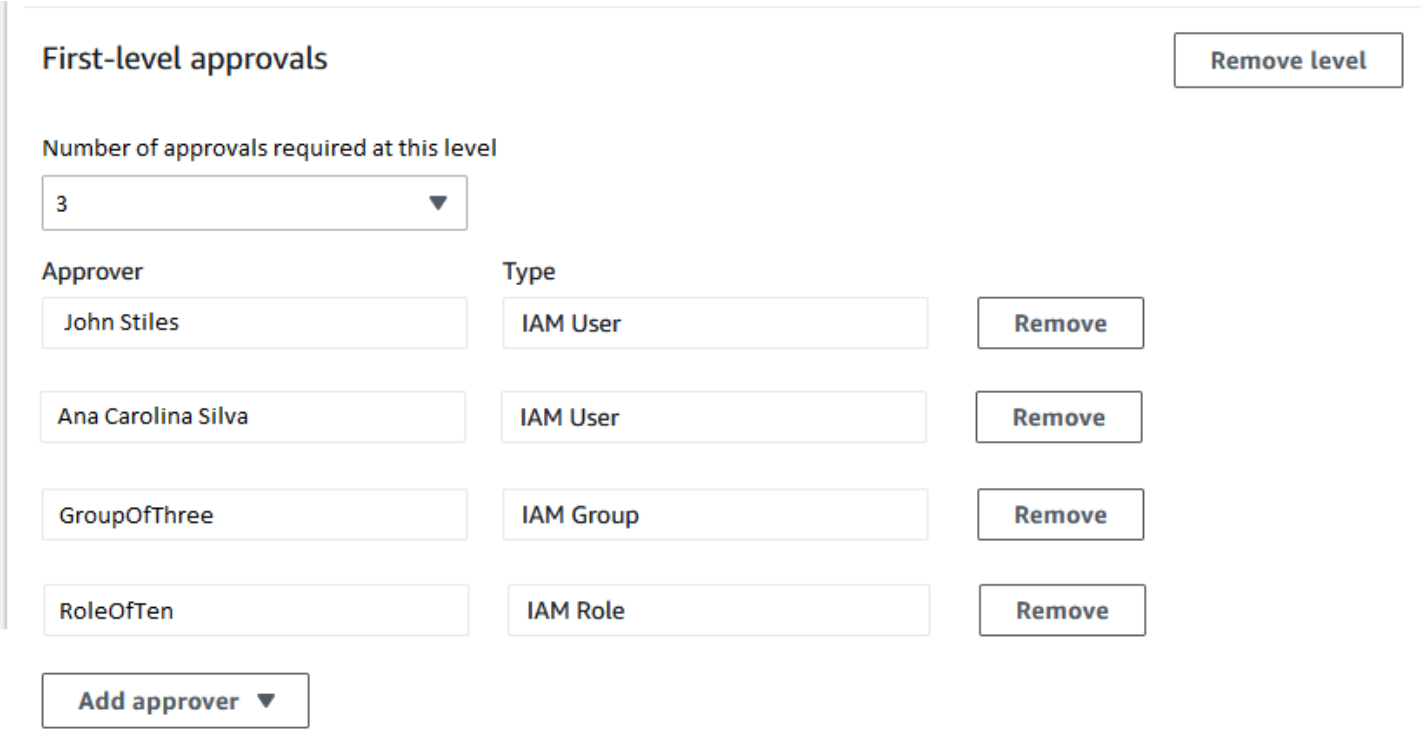
À l'aide de modèles de modification créés avant le 23 janvier 2023, une approbation doit être reçue de chaque approbateur spécifié pour que la demande de modification soit approuvée à ce niveau. Par exemple, dans la configuration du niveau d'approbation présentée dans l'image suivante, quatre approbateurs sont spécifiés. Les approbateurs spécifiés incluent deux utilisateurs (John Stiles et Ana Carolina Silva), un groupe d'utilisateurs composé de trois membres (GroupOfThree) et un rôle d'utilisateur représentant dix utilisateurs (RoleOfTen).



Pour que la demande de modification soit approuvée à ce niveau, elle doit être approuvée par John Stiles, Ana Carolina Silva, un membre du groupe `GroupOfThree` et un membre du rôle `RoleOfTen`.

À l'aide de modèles de modification créés le 23 janvier 2023 ou après cette date, les créateurs de modèles peuvent spécifier le nombre total d'approbations requises pour chaque niveau d'approbation. Ces approbations peuvent provenir de n'importe quelle combinaison d'utilisateurs, de groupes et de rôles qui ont été spécifiés en tant qu'approbateurs. Un modèle de modification peut nécessiter une seule approbation pour un niveau, mais spécifier, par exemple, deux utilisateurs individuels, deux groupes et un rôle en tant qu'approbateurs potentiels.

Par exemple, dans la zone du niveau d'approbation présentée dans l'image suivante, trois approbations sont requises. Les approbateurs spécifiés dans le modèle incluent deux utilisateurs (John Stiles et Ana Carolina Silva), un groupe d'utilisateurs composé de trois membres (`GroupOfThree`) et un rôle utilisateur représentant dix utilisateurs (`RoleOfTen`).



Si les trois utilisateurs du groupe `GroupOfThree` approuvent votre demande de modifications, elle est approuvée pour ce niveau. Il n'est pas nécessaire de recevoir l'approbation de chaque utilisateur, groupe ou rôle. Le nombre minimum d'approbations peut provenir de n'importe quelle combinaison d'approbateurs potentiels.

Lorsque votre demande de modifications est créée, des notifications sont envoyées aux abonnés du sujet Amazon SNS qui a été spécifié pour les notifications d'approbation à ce niveau. Le créateur du modèle de modifications a peut-être spécifié le sujet de notification à utiliser ou vous a autorisé à en spécifier un.

Une fois que le nombre minimum d'approbations requises est reçu à un niveau, des notifications sont envoyées aux approbateurs abonnés à la rubrique Amazon SNS pour le niveau suivant, et ainsi de suite.

Quel que soit le nombre de niveaux d'approbation et d'approbateurs spécifiés, un seul rejet d'une demande de modifications est requis pour empêcher le flux de travail du runbook pour cette demande de se produire.

Création de demandes de modifications (console)

La procédure suivante décrit comment créer une demande de modification à l'aide de la console Systems Manager.

Pour créer une demande de modification (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Change Manager.
3. Sélectionnez Créer une demande.
4. Recherchez et sélectionnez un modèle de modification à utiliser pour cette demande de modification.
5. Sélectionnez Suivant.
6. Pour Nom de la demande de modification, saisissez un nom permettant d'identifier sa fonction, **`UpdateEC2LinuxAMI-us-east-2`** par exemple.
7. Pour Runbook, sélectionnez le runbook à utiliser pour effectuer la modification demandée.

   > **Note**
   >
   > Si l'option permettant de sélectionner un runbook n'est pas disponible, le créateur du modèle de modification a spécifié le runbook à utiliser.

8. Pour Informations sur la demande de modification, utilisez Markdown pour fournir des informations supplémentaires sur la demande de modification afin d'aider les vérificateurs à décider s'ils doivent approuver ou rejeter la demande de modification. Le créateur du modèle que vous utilisez peut avoir fourni des instructions ou des questions pour répondre.

   > **Note**
   >
   > Markdown est 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.

9. Dans la section Heure de début du flux de travail, sélectionnez l'une des options suivantes :

   - Exécuter l'opération à une heure planifiée : pourHeure de début demandée, saisissez la date et l'heure auxquelles vous proposez d'exécuter le flux de travail de runbook pour cette demande. Pour Heure de fin estimée, saisissez la date et l'heure de la fin attendue du flux de travail du runbook. (Cette heure n'est qu'une estimation destinée aux vérificateurs.)

> **Tip**
>
> Sélectionnez Afficher le calendrier des modifications pour vérifier les événements de blocage pendant la durée spécifiée.

- Exécuter l'opération dès que possible après l'approbation : si la demande de modification est approuvée, le flux de travail de runbook s'exécute dès qu'une période sans restriction se présente, pendant laquelle les modifications peuvent être apportées.

10. Dans la section Change request approvals (Approbations de demande de modification), procédez de la manière suivante :

    1. Si des options Type d'approbation sont proposées, sélectionnez l'une des options suivantes :

       - Approbation automatique : le modèle de modification que vous avez sélectionné est configuré pour autoriser l'exécution automatique des demandes de modifications sans vérification par les approbateurs. Passez à l'étape 11.

         > **Note**
         >
         > Les autorisations spécifiées dans les politiques IAM, qui régissent votre utilisation de Systems Manager, ne doivent pas vous empêcher d'envoyer des demandes de modifications d'approbation automatique pour qu'elles s'exécutent automatiquement.

       - Spécification des approbateurs : vous devez ajouter un ou plusieurs utilisateurs, groupes ou rôles IAM pour vérifier et approuver cette demande de modification.

         > **Note**
         >
         > Vous pouvez choisir de spécifier des vérificateurs même si les autorisations spécifiées dans les politiques IAM qui régissent votre utilisation du Systems Manager vous autorisent à exécuter des demandes de modifications d'approbation automatique.

    2. Choisissez Ajouter un approbateur, puis sélectionnez un ou plusieurs utilisateurs, groupes ou rôles AWS Identity and Access Management (IAM) dans la liste des réviseurs disponibles.

> **Note**
>
> Un ou plusieurs approbateurs peuvent déjà être spécifiés. Cela signifie que les approbateurs obligatoires sont déjà spécifiés dans le modèle de modification que vous avez sélectionné. Ces approbateurs ne peuvent pas être supprimés de la demande. Si le bouton Ajouter un approbateur n'est pas activé, le modèle que vous avez choisi n'autorise pas l'ajout de vérificateurs supplémentaires aux demandes.

Pour obtenir des informations sur les approbations de demande de modifications, consultez À propos des approbations de demandes de modification.

3. Sous Rubrique SNS pour notifier les approbateurs, sélectionnez l'une des options suivantes pour spécifier la rubrique Amazon SNS de votre compte à utiliser pour envoyer des notifications aux approbateurs que vous ajoutez à cette demande de modification.

> **Note**
>
> Si l'option permettant de spécifier une rubrique Amazon SNS n'est pas disponible, la rubrique Amazon SNS à utiliser est déjà spécifiée dans le modèle de modification que vous avez sélectionné.

- Saisir un Amazon Resource Name (ARN) SNS : pour ARN de rubrique, saisissez l'ARN d'une rubrique Amazon SNS existante. Cette rubrique peut se trouver dans n'importe quel compte de votre organisation.
- Sélectionner une rubrique SNS existante : pour Rubrique de notification cible, sélectionnez l'ARN d'une rubrique Amazon SNS existante dans votre compte actuel. (Cette option n'est pas disponible si vous n'avez pas encore créé de rubrique Amazon SNS dans votre Compte AWS et Région AWS.)

> **Note**
>
> La rubrique Amazon SNS que vous sélectionnez doit être configurée pour spécifier les notifications qu'elle envoie, ainsi que les abonnés auxquels elles sont envoyées. Sa politique d'accès doit également octroyer des autorisations à Systems Manager de sorte que Change Manager puisse envoyer des notifications. Pour plus d'informations,

consultez Configuration des rubriques Amazon SNS pour les notifications Change Manager.

4. Sélectionnez Ajouter une notification.

11. Sélectionnez Suivant.
12. Pour IAM role (Rôle IAM), sélectionnez dans votre compte actuel un rôle IAM qui dispose des autorisations nécessaires pour exécuter les runbooks spécifiés pour cette demande de modification.

    Ce rôle est également appelé fonction du service, ou rôle d'assumer, pour Automation. Pour plus d'informations sur ce rôle, consultez Configuration d'Automation.
13. Dans la section Deployment location (Emplacement de déploiement), sélectionnez l'une des options suivantes :

> **Note**
>
> Si vous l'utilisez Change Manager avec une Compte AWS seule personne et non avec une organisation configurée dans AWS Organizations, il n'est pas nécessaire de spécifier un lieu de déploiement.

- Appliquer la modification à ce compte : le flux de travail de runbook s'exécute uniquement dans le compte actuel. Pour une organisation, cela désigne le compte d'administrateur délégué.
- Appliquer une modification à plusieurs unités organisationnelles (UO) : procédez comme suit :
  1. Pour Accounts and organizational units (OUs) (Comptes et unités organisationnelles (UO)), saisissez l'ID d'un compte membre dans votre organisation, au format **123456789012**, ou l'ID d'une unité organisationnelle, au format **o-o96EXAMPLE**.
  2. (Facultatif) Pour Execution role name (Nom du rôle d'exécution), saisissez le nom du rôle IAM dans le compte cible ou de l'UO qui dispose des autorisations nécessaires pour exécuter les runbooks spécifiés pour cette demande de modification. Tous les comptes de l'UO que vous spécifiez doivent utiliser le même nom pour ce rôle.
  3. (Facultatif) Sélectionnez Add another target location (Ajouter un emplacement cible) pour chaque compte ou UO que vous voulez spécifier, et répétez les étapes a et b.
  4. Pour Target Région AWS, sélectionnez la région dans laquelle effectuer le changement, par exemple `Ohio (us-east-2)` pour la région USA Est (Ohio).

5. Développez Rate control (Contrôle de débit).

   Pour Concurrency (Concomitance), saisissez un nombre, puis, dans la liste, sélectionnez si cela représente le nombre ou le pourcentage de comptes dans lesquels le flux de travail de runbook peut s'exécuter simultanément.

   Pour Error threshold (Seuil d'erreur), saisissez un nombre, puis, dans la liste, sélectionnez si cela représente le nombre ou le pourcentage de comptes dans lesquels le flux de travail de runbook peut échouer avant l'arrêt de l'opération.

14. Dans la section Deployment targets (Cibles de déploiement), procédez comme suit :

    1. Sélectionnez l'une des méthodes suivantes :

       - Ressource unique : la modification doit être apportée pour une seule ressource. Par exemple, un seul nœud ou une seule AMI (Amazon Machine Image), en fonction de l'opération définie dans les runbooks pour cette demande de modification.
       - Plusieurs ressources : pour Parameter (Paramètre), sélectionnez parmi les paramètres disponibles dans les runbooks pour cette demande de modification. Cette sélection reflète le type de ressource mise à jour.

         Par exemple, si le runbook pour cette demande de modification est `AWS-RetartEC2Instance`, vous pouvez choisir `InstanceId`, puis définir les instances qui sont mises à jour en sélectionnant l'une des options suivantes :

         - Spécifier des balises : saisissez une paire clé-valeur avec laquelle toutes les ressources à mettre à jour sont balisées.
         - Choisir un groupe de ressources : sélectionnez le nom du groupe de ressources auquel appartiennent toutes les ressources à mettre à jour.
         - Spécifier les valeurs des paramètres : identifiez les ressources à mettre à jour dans la section Runbook parameters (Paramètres du Runbook).
         - Target all instances (Cibler toutes les instances) : apportez la modification sur tous les nœuds gérés des emplacements cibles.

    2. Si vous avez choisi Plusieurs ressources, développez Contrôle de débit.

       Pour Concurrency (Concomitance), saisissez un nombre, puis, dans la liste, sélectionnez si cela représente le nombre ou le pourcentage de comptes que le flux de travail de runbook peut mettre à jour simultanément.

Pour Error threshold (Seuil d'erreur), saisissez un nombre, puis, dans la liste, sélectionnez si cela représente le nombre ou le pourcentage de cibles dans lesquelles la mise à jour peut échouer avant l'arrêt de l'opération.

15. Si vous avez choisi Spécifier les valeurs des paramètres pour mettre à jour plusieurs ressources à l'étape précédente : dans la section Paramètres du Runbook, spécifiez des valeurs pour les paramètres d'entrée requis. Les valeurs de paramètre que vous devez fournir sont basées sur le contenu des runbooks Automation associés au modèle de modification que vous avez choisi.

    Par exemple, si le modèle de modification utilise le `AWS-RetartEC2Instance` runbook, vous devez saisir un ou plusieurs ID d'instance pour le InstanceIdparamètre. Sinon, sélectionnez Show interactive instance picker (Afficher le sélecteur d'instance interactif) et sélectionnez les instances disponibles une par une.

16. Sélectionnez Suivant.
17. Sur la page Review and submit (Vérifier et envoyer), re-vérifiez les ressources et les options que vous avez spécifiées pour cette demande de modification.

    Sélectionnez le bouton Edit (Modifier) pour la ou les sections auxquelles vous voulez apporter des modifications.

    Lorsque vous êtes satisfait des détails de la demande de modification, sélectionnez Submit for approval (Envoyer pour approbation).

Si une rubrique Amazon SNS a été spécifiée pour le modèle des modifications que vous avez choisi pour la demande, des notifications sont envoyées lorsque la demande est rejetée ou approuvée. Si vous ne recevez pas de notifications pour la demande, vous pouvez revenir à Change Manager pour vérifier le statut de votre demande.

Création de demandes de modifications (AWS CLI)

Vous pouvez créer une demande de modification à l'aide de AWS Command Line Interface (AWS CLI) en spécifiant les options et les paramètres de la demande de modification dans un fichier JSON et en utilisant l'`--cli-input-json`option pour l'inclure dans votre commande.

Pour créer une demande de modification (AWS CLI)

1. Installez et configurez le AWS CLI ou le AWS Tools for PowerShell, si ce n'est pas déjà fait.

Pour plus d'informations, consultez la section Installation ou mise à jour de la version la plus récente de l' AWS CLI et Installation d' AWS Tools for PowerShell.

2. Créez un fichier JSON sur votre ordinateur local avec un nom tel que `MyChangeRequest.json`, puis collez le contenu suivant dans le fichier.

   Remplacez les *espaces réservés* par des valeurs pour votre demande de modification.

   > **Note**
   >
   > Cet exemple JSON crée une demande de modification en utilisant le modèle de modification `AWS-HelloWorldChangeTemplate` et le runbook `AWS-HelloWorld`. Pour vous aider à adapter cet exemple à vos propres demandes de modifications, consultez StartChangeRequestExecution dans la Référence d'API AWS Systems Manager pour obtenir des informations sur les paramètres disponibles
   > Pour obtenir des informations sur les approbations de demande de modifications, consultez À propos des approbations de demandes de modification.

```
{
    "ChangeRequestName": "MyChangeRequest",
    "DocumentName": "AWS-HelloWorldChangeTemplate",
    "DocumentVersion": "$DEFAULT",
    "ScheduledTime": "2021-12-30T03:00:00",
    "ScheduledEndTime": "2021-12-30T03:05:00",
    "Tags": [
        {
            "Key": "Purpose",
            "Value": "Testing"
        }
    ],
    "Parameters": {
        "Approver": [
            "JohnDoe"
        ],
        "ApproverType": [
            "IamUser"
        ],
        "ApproverSnsTopicArn": [
            "arn:aws:sns:us-east-2:123456789012:MyNotificationTopic"
```

```
        ]
    },
    "Runbooks": [
        {
            "DocumentName": "AWS-HelloWorld",
            "DocumentVersion": "1",
            "MaxConcurrency": "1",
            "MaxErrors": "1",
            "Parameters": {
                "AutomationAssumeRole": [
                    "arn:aws:iam::123456789012:role/MyChangeManagerAssumeRole"
                ]
            }
        }
    ],
    "ChangeDetails": "### Document Name: HelloWorldChangeTemplate\n\n## What does
 this document do?\nThis change template demonstrates the feature set available
 for creating change templates for Change Manager. This template starts a Runbook
 workflow for the Automation document called AWS-HelloWorld.\n\n## Input Parameters
\n* ApproverSnsTopicArn: (Required) Amazon Simple Notification Service ARN for
 approvers.\n* Approver: (Required) The name of the approver to send this request
 to.\n* ApproverType: (Required) The type of reviewer.\n  * Allowed Values:
 IamUser, IamGroup, IamRole, SSOGroup, SSOUser\n\n## Output Parameters\nThis
 document has no outputs \n"
}
```

3. Dans le répertoire où vous avez créé le fichier JSON, exécutez la commande suivante.

```
aws ssm start-change-request-execution --cli-input-json file://MyChangeRequest.json
```

Le système retourne des informations telles que les suivantes.

```
{
    "AutomationExecutionId": "b3c1357a-5756-4839-8617-2d2a4EXAMPLE"
}
```

Vérifier et approuver ou rejeter les demandes de modifications

Si vous êtes désigné comme réviseur pour une demande de modification dans une fonctionnalité de Change Manager AWS Systems Manager, vous êtes averti par le biais d'une rubrique Amazon

Simple Notification Service (Amazon SNS) lorsqu'une nouvelle demande de modification est en attente de révision.

> **Note**
>
> Cette fonctionnalité dépend du fait qu'un Amazon SNS a été spécifié, ou non, dans le modèle de modification pour l'envoi de notifications de vérification. Pour plus d'informations, consultez Configuration des rubriques Amazon SNS pour les notifications Change Manager.

Pour consulter la demande de modification, vous pouvez suivre le lien figurant dans votre notification, ou vous connecter AWS Management Console directement au et suivre les étapes de cette procédure.

> **Note**
>
> Si une rubrique Amazon SNS est affectée aux vérificateurs dans un modèle de modification, des notifications sont envoyées aux abonnés de la rubrique lorsque la demande de modification change de statut.
> Pour obtenir des informations sur les approbations de demande de modifications, consultez À propos des approbations de demandes de modification.

Vérifier et approuver ou rejeter les demandes de modifications (console)

Les procédures suivantes décrivent comment utiliser la console Systems Manager pour examiner, approuver ou rejeter une demande de modification.

Pour la vérification, l'approbation ou le rejet d'une seule demande de modification

1. Ouvrez le lien contenu dans la notification par e-mail que vous avez reçue et connectez-vous au AWS Management Console, qui vous redirige vers la demande de modification à examiner.
2. Sur la page récapitulative, vérifiez le contenu proposé de la demande de modification.

   Pour approuver la demande de modification, sélectionnez Approve (Approuver). Dans la boîte de dialogue, fournissez d'éventuels commentaires à ajouter pour cette approbation, puis sélectionnez Approve (Approuver). Le flux de travail de runbook représenté par cette demande commence à s'exécuter, soit lorsqu'il est planifié, soit dès que les modifications ne sont pas bloquées par aucune restriction.

-ou-

Pour rejeter la demande de modification, sélectionnez Reject (Rejeter). Dans la boîte de dialogue, fournissez d'éventuels commentaires à ajouter pour ce rejet, puis sélectionnez Reject (Rejeter).

Pour la vérification, l'approbation ou le rejet de plusieurs demandes de modification

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Change Manager.
3. Cliquez sur l'onglet Approvals (Approbations).
4. (Facultatif) Consultez les détails des demandes en attente d'approbation en choisissant le nom de chaque demande, puis revenez à l'onglet Approvals (Approbations).
5. Cochez la case correspondant à demande de modification que vous souhaitez approuver.

   -ou-

   Cochez la case correspondant à demande de modification que vous souhaitez rejeter.
6. Dans la boîte de dialogue, saisissez d'éventuels commentaires à ajouter pour l'approbation ou le rejet.
7. Selon que vous décision d'approuver ou de rejeter les demandes de modification sélectionnées, choisissez Approver (Approuver) ou Reject (Rejeter).

Vérifier et approuver ou rejeter les demandes de modifications (ligne de commande)

La procédure suivante décrit comment utiliser le AWS Command Line Interface (AWS CLI) (sous Linux ou Windows) pour examiner et approuver ou rejeter une demande de modification. macOS

Pour vérifier et approuver ou rejeter une demande de modification

1. Installez et configurez le AWS Command Line Interface (AWS CLI), si ce n'est pas déjà fait.

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l' AWS CLI.
2. Créez un fichier JSON sur votre machine locale qui spécifie les paramètres de votre AWS CLI appel.

```
{
  "OpsItemFilters":
  [
    {
      "Key": "OpsItemType",
      "Values": ["/aws/changerequest"],
      "Operator": "Equal"
    }
  ],
  "MaxResults": number
}
```

Vous pouvez filtrer les résultats pour un approbateur spécifique en spécifiant l'Amazon Resource Name (ARN) de cet approbateur dans le fichier JSON. Voici un exemple.

```
{
  "OpsItemFilters":
  [
    {
      "Key": "OpsItemType",
      "Values": ["/aws/changerequest"],
      "Operator": "Equal"
    },
    {
      "Key": "ChangeRequestByApproverArn",
      "Values": ["arn:aws:iam::account-id:user/user-name"],
      "Operator": "Equal"
    }
  ],
  "MaxResults": number
}
```

3. Exécutez la commande suivante pour afficher le nombre maximal de demandes de modifications que vous avez spécifié dans le fichier JSON.

   Linux & macOS

   ```
   aws ssm describe-ops-items \
   --cli-input-json file://filename.json
   ```

Windows

```
aws ssm describe-ops-items ^
--cli-input-json file://filename.json
```

4. Exécutez la commande suivante pour approuver ou rejeter une demande de modification.

   Linux & macOS

   ```
   aws ssm send-automation-signal \
       --automation-execution-id ID \
       --signal-type Approve_or_Reject \
       --payload Comment="message"
   ```

   Windows

   ```
   aws ssm send-automation-signal ^
   --automation-execution-id ID ^
       --signal-type Approve_or_Reject ^
       --payload Comment="message"
   ```

   Si une rubrique Amazon SNS a été spécifiée pour le modèle des modifications que vous avez choisi pour la demande, des notifications sont envoyées lorsque la demande est rejetée ou approuvée. Si vous ne recevez pas de notifications pour la demande, vous pouvez revenir à Change Manager pour vérifier le statut de votre demande. Pour plus d'informations sur les autres options disponibles lors de l'utilisation de cette commande, consultez send-automation-signal dans la section AWS Systems Manager de la Référence des commandes AWS CLI .

Vérifier les détails, les tâches et les échéances d'une demande de modification (console)

Vous pouvez afficher des informations sur une demande de modification, y compris les demandes pour lesquelles des modifications ont déjà été traitées, dans le tableau de bord de Change Manager, une fonctionnalité de AWS Systems Manager. Ces détails incluent un lien vers l'opération Automation, qui exécute les runbooks effectuant la modification. Un ID d'exécution Automation est généré lors de la création de la demande, mais le processus ne s'exécute pas tant que toutes les

approbations n'ont pas été données et que plus aucune restriction n'est en place pour bloquer la modification.

Pour vérifier les détails, les tâches et les échéances d'une demande de modification (console)

1. Dans le panneau de navigation, sélectionnez Change Manager.
2. Sélectionnez l'onglet Requests (Demandes).
3. Dans la section Change requests (Demandes de modifications), recherchez la demande de modification à vérifier.

   Vous pouvez utiliser les options Create date range (Créer une plage de dates) pour limiter les résultats à une période spécifique.

   Vous pouvez filtrer les demandes sur la base des propriétés suivantes :

   - `Status`
   - `Request ID`
   - `Approver`
   - `Requester`

   Par exemple, pour afficher des détails sur toutes les demandes de modifications qui se sont achevées avec succès au cours des dernières 24 heures, procédez comme suit :

   1. Pour Créer une plage de dates, sélectionnez 1d.
   2. Dans le champ de recherche, sélectionnez Status, CompletedWithSuccess.
   3. Dans les résultats, sélectionnez le nom de la demande de modification terminée avec succès pour vérifier les résultats.
4. Affichez les informations sur la demande de modification dans les onglets suivants :

   - Détails de la demande : affiche les détails de base sur la demande de modification, notamment le demandeur, le modèle de modification et les runbooks Automation sélectionnés pour la modification. Vous pouvez également suivre un lien vers les détails de l'opération d'automatisation et consulter les informations relatives aux paramètres du runbook spécifiés dans la demande, aux CloudWatch alarmes Amazon attribuées à la demande de modification, ainsi qu'aux approbations et commentaires fournis pour la demande.

- Tâche : affiche des informations sur la tâche dans la modification, notamment le statut de la tâche pour les demandes de modifications terminées, les ressources ciblées, les étapes des runbooks Automation associés, ainsi que les détails de concomitance et de seuils d'erreur.
- Chronologie : affiche un récapitulatif de tous les événements associés à la demande de modification, répertoriés par date et heure. Le récapitulatif indique la date de création de la demande, les actions des approbateurs affectés, une note sur la date de l'exécution planifiée des demandes de modifications approuvées, les détails du flux de travail de runbook et les changements de statut pour le processus de modification dans son ensemble et chaque étape du runbook.
- Associated events (Événements associés) : affichez des détails vérifiables sur les demandes de modification enregistrées dans AWS CloudTrail Lake. Les détails incluent les actions d'API qui ont été exécutées, les paramètres de demande inclus pour ces actions, le compte utilisateur qui a exécuté l'action, les ressources mises à jour au cours du processus, etc.

  Lorsque vous activez CloudTrail le suivi des événements de CloudTrail Lake, Lake crée un magasin de données d'événements pour les événements liés à vos demandes de modification. Les détails de l'événement sont disponibles pour le compte ou l'organisation où la demande de modification a été exécutée. Vous pouvez activer le suivi des événements CloudTrail Lake à partir de toute demande de modification concernant votre compte ou votre organisation. Pour plus d'informations sur l'activation de l'intégration de CloudTrail Lake et la création d'un magasin de données d'événements, consultezSurveillance de vos événements de demande de modification.

  > **Note**
  >
  > L'utilisation du CloudTrail lac est payante. Pour de plus amples informations, veuillez consulter Tarification AWS CloudTrail.

Affichage du nombre agrégé de demandes de modifications (ligne de commande)

Vous pouvez afficher le nombre agrégé de demandes de modifications dans Change Manager, une fonctionnalité de AWS Systems Manager, en utilisant l'opération d'API GetOpsSummary. Cette opération d'API peut renvoyer des nombres pour un Compte AWS unique dans une Région AWS unique ou pour plusieurs comptes et plusieurs régions.

> **Note**
>
> Pour afficher le nombre agrégé de demandes de modifications pour plusieurs Comptes AWS et plusieurs Régions AWS, vous devez paramétrer et configurer une synchronisation de données de ressources. Pour de plus amples informations, veuillez consulter Configuration de la synchronisation de données de ressource pour Inventory.

La procédure suivante décrit comment utiliser la AWS Command Line Interface (AWS CLI) (sur Linux,macOS ou Windows) pour afficher le nombre agrégé de demandes de modifications.

Pour afficher les nombres agrégés de demandes de modifications

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS Command Line Interface (AWS CLI).

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l'AWS CLI.
2. Exécutez une des commandes suivantes :

   Compte et région uniques

   Cette commande renvoie un nombre de toutes les demandes de modifications pour le Compte AWS et la Région AWS pour lesquels votre session AWS CLI est configurée.

   Linux & macOS

   ```
   aws ssm get-ops-summary \
   --filters Key=AWS:OpsItem.OpsItemType,Values="/aws/changerequests",Type=Equal \
   --aggregators AggregatorType=count,AttributeName=Status,TypeName=AWS:OpsItem
   ```

   Windows

   ```
   aws ssm get-ops-summary ^
   --filters Key=AWS:OpsItem.OpsItemType,Values="/aws/changerequests",Type=Equal ^
   --aggregators AggregatorType=count,AttributeName=Status,TypeName=AWS:OpsItem
   ```

   L'appel renvoie des informations telles que les suivantes.

   ```
   {
   ```

```
    "Entities": [
        {
            "Data": {
                "AWS:OpsItem": {
                    "Content": [
                        {
                            "Count": "38",
                            "Status": "Open"
                        }
                    ]
                }
            }
        }
    ]
}
```

Comptes et régions multiples

Cette commande renvoie un nombre de toutes les demandes de modifications pour le Comptes AWS et la Régions AWS spécifiés dans la synchronisation des données de ressource.

Linux & macOS

```
aws ssm get-ops-summary \
    --sync-name resource_data_sync_name \
    --filters Key=AWS:OpsItem.OpsItemType,Values="/aws/
changerequests",Type=Equal \
    --aggregators AggregatorType=count,AttributeName=Status,TypeName=AWS:OpsItem
```

Windows

```
aws ssm get-ops-summary ^
    --sync-name resource_data_sync_name ^
    --filters Key=AWS:OpsItem.OpsItemType,Values="/aws/
changerequests",Type=Equal ^
    --aggregators AggregatorType=count,AttributeName=Status,TypeName=AWS:OpsItem
```

L'appel renvoie des informations telles que les suivantes.

```
{
    "Entities": [
```

```
        {
            "Data": {
                "AWS:OpsItem": {
                    "Content": [
                        {
                            "Count": "43",
                            "Status": "Open"
                        },
                        {
                            "Count": "2",
                            "Status": "Resolved"
                        }
                    ]
                }
            }
        }
    ]
}
```

Plusieurs comptes et une région spécifique

Cette commande renvoie un nombre de toutes les demandes de modifications pour les Comptes AWS spécifiés dans la synchronisation des données de ressources. Elle renvoie toutefois uniquement les données de la région spécifiée dans la commande.

Linux & macOS

```
aws ssm get-ops-summary \
    --sync-name resource_data_sync_name \
    --filters Key=AWS:OpsItem.SourceRegion,Values='Region',Type=Equal
 Key=AWS:OpsItem.OpsItemType,Values="/aws/changerequests",Type=Equal \
    --aggregators AggregatorType=count,AttributeName=Status,TypeName=AWS:OpsItem
```

Windows

```
aws ssm get-ops-summary ^
    --sync-name resource_data_sync_name ^
    --filters Key=AWS:OpsItem.SourceRegion,Values='Region',Type=Equal
 Key=AWS:OpsItem.OpsItemType,Values="/aws/changerequests",Type=Equal ^
    --aggregators AggregatorType=count,AttributeName=Status,TypeName=AWS:OpsItem
```

Comptes et régions multiples avec une sortie groupée par région

Cette commande renvoie un nombre de toutes les demandes de modifications pour les Comptes AWS et les Régions AWS spécifiés dans la synchronisation des données de ressources. La sortie affiche les informations de comptage par région.

Linux & macOS

```
aws ssm get-ops-summary \
    --sync-name resource_data_sync_name \
    --filters Key=AWS:OpsItem.OpsItemType,Values="/aws/
changerequests",Type=Equal \
    --aggregators
 '[{"AggregatorType":"count","TypeName":"AWS:OpsItem","AttributeName":"Status","Aggregat
[{"AggregatorType":"count","TypeName":"AWS:OpsItem","AttributeName":"SourceRegion"}]}]'
```

Windows

```
aws ssm get-ops-summary ^
    --sync-name resource_data_sync_name ^
    --filters Key=AWS:OpsItem.OpsItemType,Values="/aws/
changerequests",Type=Equal ^
    --aggregators
 '[{"AggregatorType":"count","TypeName":"AWS:OpsItem","AttributeName":"Status","Aggregat
[{"AggregatorType":"count","TypeName":"AWS:OpsItem","AttributeName":"SourceRegion"}]}]'
```

L'appel renvoie des informations telles que les suivantes.

```
{
        "Entities": [
            {
                "Data": {
                    "AWS:OpsItem": {
                        "Content": [
                            {
                                "Count": "38",
                                "SourceRegion": "us-east-1",
                                "Status": "Open"
                            },
```

```
                        {
                            "Count": "4",
                            "SourceRegion": "us-east-2",
                            "Status": "Open"
                        },
                        {
                            "Count": "1",
                            "SourceRegion": "us-west-1",
                            "Status": "Open"
                        },
                        {
                            "Count": "2",
                            "SourceRegion": "us-east-2",
                            "Status": "Resolved"
                        }
                    ]
                }
            }
        ]
    }
```

Comptes et régions multiples avec une sortie groupée par comptes et par régions

Cette commande renvoie un nombre de toutes les demandes de modifications pour les Comptes AWS et les Régions AWS spécifiés dans la synchronisation des données de ressources. La sortie regroupe les informations de nombres, par comptes et par régions.

Linux & macOS

```
aws ssm get-ops-summary \
    --sync-name resource_data_sync_name \
    --filters Key=AWS:OpsItem.OpsItemType,Values="/aws/
changerequests",Type=Equal \
    --aggregators
 '[{"AggregatorType":"count","TypeName":"AWS:OpsItem","AttributeName":"Status","Aggregat
[{"AggregatorType":"count","TypeName":"AWS:OpsItem","AttributeName":"SourceAccountId","A
[{"AggregatorType":"count","TypeName":"AWS:OpsItem","AttributeName":"SourceRegion"}]}]}]
```

Windows

```
aws ssm get-ops-summary ^
```

```
    --sync-name resource_data_sync_name ^
    --filters Key=AWS:OpsItem.OpsItemType,Values="/aws/
changerequests",Type=Equal ^
    --aggregators
 '[{"AggregatorType":"count","TypeName":"AWS:OpsItem","AttributeName":"Status","Aggregat
[{"AggregatorType":"count","TypeName":"AWS:OpsItem","AttributeName":"SourceAccountId","A
[{"AggregatorType":"count","TypeName":"AWS:OpsItem","AttributeName":"SourceRegion"}]}]}]
```

L'appel renvoie des informations telles que les suivantes.

```
{
    "Entities": [
        {
            "Data": {
                "AWS:OpsItem": {
                    "Content": [
                        {
                            "Count": "38",
                            "SourceAccountId": "123456789012",
                            "SourceRegion": "us-east-1",
                            "Status": "Open"
                        },
                        {
                            "Count": "4",
                            "SourceAccountId": "111122223333",
                            "SourceRegion": "us-east-2",
                            "Status": "Open"
                        },
                        {
                            "Count": "1",
                            "SourceAccountId": "111122223333",
                            "SourceRegion": "us-west-1",
                            "Status": "Open"
                        },
                        {
                            "Count": "2",
                            "SourceAccountId": "444455556666",
                            "SourceRegion": "us-east-2",
                            "Status": "Resolved"
                        },
                        {
                            "Count": "1",
```

```
                            "SourceAccountId": "222222222222",
                            "SourceRegion": "us-east-1",
                            "Status": "Open"
                        }
                    ]
                }
            }
        }
    ]
}
```

Audit et journalisation de l'activité de Change Manager

Vous pouvez auditer l'activité de Change Manager, une fonctionnalité de AWS Systems Manager, en utilisant les alarmes Amazon CloudWatch et de la AWS CloudTrail.

Pour de plus amples informations sur les options d'audit et de journalisation de Systems Manager, veuillez consulter Surveillance AWS Systems Manager.

Auditez l'activité de Change Manager en utilisant les alarmes CloudWatch

Vous pouvez configurer et affecter une alarme CloudWatch à un modèle de modification. Si les conditions définies dans l'alarme sont remplies, les actions spécifiées pour l'alarme sont effectuées. Dans la configuration de l'alarme, vous pouvez spécifier une rubrique Amazon Simple Notification Service (Amazon SNS), qui enverra une notification lorsqu'une condition d'alarme sera remplie.

Pour de plus amples informations sur la création d'un modèle Change Manager, veuillez consulter Utilisation des modèles de modification.

Pour plus d'informations sur la création d'alarmes CloudWatch, consultez Utilisation des alarmes CloudWatch dans le Guide de l'utilisateur Amazon CloudWatch.

Auditez l'activité de Change Manager en utilisant CloudTrail

CloudTrail capture les appels d'API effectués dans la console Systems Manager, la AWS Command Line Interface (AWS CLI) et le kit SDK Systems Manager. Vous pouvez afficher les informations dans la console CloudTrail ou dans un compartiment Amazon Simple Storage Service (Amazon S3) où elles sont stockées. Un seul compartiment est utilisé pour tous les journaux CloudTrail de votre compte.

Les journaux d'actions Change Manager montrent la création de document de modèle de modification, les approbations et les refus des modèles de modifications et des demandes de modifications, l'activité générée par les runbooks Automation, etc. 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' AWS Systems Manager API avec AWS CloudTrail.

Résolution des problèmes de Change Manager

Utilisez les informations suivantes pour essayer de résoudre les problèmes liés à Change Manager, une des fonctionnalités de AWS Systems Manager.

Rubriques

- Erreur « Groupe{GUID}introuvable » lors d'approbations de demandes de modifications en utilisant Active Directory (groupes

Erreur « Groupe*{GUID}*introuvable » lors d'approbations de demandes de modifications en utilisant Active Directory (groupes

Problème : quand AWS IAM Identity Center (IAM Identity Center) est utilisé pour la gestion des identités des utilisateurs, un membre d'un groupe Active Directory auquel les autorisations d'approbation sont accordées dans Change Manager reçoit un message d'erreur « non autorisé » ou « groupe introuvable ».

- Solution : lorsque vous sélectionnez des groupes Active Directory dans IAM Identity Center pour accéder à la AWS Management Console, le système planifie une synchronisation périodique qui copie les informations de ces groupes Active Directory vers IAM Identity Center. Ce processus doit se terminer avant que les utilisateurs autorisés via l'appartenance à un groupe Active Directory puissent approuver une demande. Pour plus d'informations, consultez Connexion à votre annuaire Microsoft AD dans le Guide de l'utilisateur AWS IAM Identity Center.

AWS Systems Manager Automatisation

Automation, une fonctionnalité de AWS Systems Manager, simplifie les tâches courantes de maintenance, de déploiement et de correction pour des Services AWS comme Amazon Elastic Compute Cloud (Amazon EC2), Amazon Relational Database Service (Amazon RDS), Amazon Redshift, Amazon Simple Storage Service (Amazon S3) et bien d'autres. Pour vos premiers pas

dans l'automatisaton, ouvrez Systems Manager console. Dans le panneau de navigation de gauche, sélectionnez Automation (Automatisation).

Automation vous aide à créer des solutions automatisées pour déployer, configurer et gérer des ressources AWS à grande échelle. Avec Automation, vous disposez d'un contrôle granulaire sur la simultanéité de vos automatisations. Cela signifie que vous pouvez spécifier le nombre de ressources à cibler simultanément et le nombre d'erreurs pouvant se produire avant l'arrêt d'une automatisation.

Pour vous aider à faire vos premiers pas avec Automation, AWS développe et gère plusieurs runbooks prédéfinis. Selon votre cas d'utilisation, vous pouvez utiliser ces runbooks prédéfinis qui effectuent diverses tâches ou créer vos propres runbooks personnalisés qui pourraient mieux répondre à vos besoins. Pour contrôler la progression et le statut de vos automatisations, vous pouvez utiliser la console Systems Manager Automation ou l'outil de ligne de commande de votre choix. L'automatisation s'intègre également EventBridge à Amazon pour vous aider à créer une architecture axée sur les événements à grande échelle.

Comment mon organisation peut-elle tirer parti d'Automation ?

Automation offre les avantages suivants :

- Prise en charge des scripting dans le contenu des runbooks

  À l'aide de `aws:executeScript` cette action, vous pouvez exécuter du Python personnalisé et des PowerShell fonctions directement à partir de vos runbooks. Cela vous offre une plus grande flexibilité dans la création de vos runbooks personnalisés, car vous pouvez effectuer diverses tâches que d'autres actions d'Automation ne prennent pas en charge. Vous avez également un meilleur contrôle sur la logique du runbook. Pour voir un exemple d'utilisation de cette action et en quoi elle peut aider à améliorer une solution automatisée existante, consultez Création de runbooks Automation.
- Exécuter des automatisations sur plusieurs Comptes AWS et Régions AWS depuis un emplacement centralisé

  Les administrateurs peuvent exécuter des automatisations sur les ressources de plusieurs comptes et régions à partir de la console Systems Manager.
- Amélioration de la sécurité opérationnelle

  Les administrateurs disposent d'un emplacement central pour accorder et révoquer l'accès aux runbooks. En utilisant uniquement les politiques AWS Identity and Access Management (IAM),

vous pouvez contrôler quels utilisateurs ou groupes au sein de votre organisation peuvent utiliser Automation et à quels runbooks ils peuvent accéder.

- Automatiser des tâches informatiques courantes

  L'automatisation des tâches courantes peut aider à améliorer l'efficacité opérationnelle, à appliquer les normes organisationnelles et à réduire les erreurs des opérateurs. Par exemple, vous pouvez utiliser le runbook `AWS-UpdateCloudFormationStackWithApproval` pour mettre à jour les ressources déployées à l'aide d'un modèle AWS CloudFormation. La mise à jour applique un nouveau modèle. Vous pouvez configurer Automation pour demander l'approbation par un ou plusieurs utilisateurs avant le début de la mise à jour.

- Exécutez simultanément des tâches perturbatrices en toute sécurité

  Automation inclut des fonctionnalités, telles que les contrôles de débit, qui vous permettent de contrôler le déploiement d'une automatisation dans votre flotte en spécifiant une valeur de simultanéité et un seuil d'erreur. Pour plus d'informations sur l'utilisation des contrôles du débit, consultez Exécution des automatisations à grande échelle.

- Rationaliser les tâches complexes

  Automation fournit des runbooks prédéfinis qui rationalisent les tâches complexes et chronophages telles que la création d'Amazon Machine Images (AMIs) finales. Par exemple, vous pouvez utiliser les runbooks `AWS-UpdateLinuxAmi` et `AWS-UpdateWindowsAmi` pour créer des AMIs finales à partir d'une source AMI. À l'aide de ces runbooks, 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 logiciels spécifiques. Pour des exemples d'utilisation de ces runbooks, consultez Didacticiels.

- Définir les contraintes pour les entrées

  Vous pouvez définir des contraintes dans des runbooks personnalisés pour limiter les valeurs qu'Automation acceptera pour un paramètre d'entrée particulier. Par exemple, `allowedPattern` n'acceptera que les valeurs d'un paramètre d'entrée correspondant à l'expression régulière que vous définissez. Si vous spécifiez `allowedValues` pour un paramètre d'entrée, seules les valeurs que vous avez spécifiées dans le runbook sont acceptées.

- Résultat de l'action d'automatisation des journaux vers Amazon CloudWatch Logs

  Pour répondre aux exigences opérationnelles ou de sécurité de votre organisation, il vous faudra peut-être fournir un registre des scripts exécutés pendant un runbook. Avec CloudWatch Logs, vous pouvez surveiller, stocker et accéder à des fichiers journaux provenant de différents types

de fichiersServices AWS. Vous pouvez envoyer le résultat de l'`aws:executeScript`action à un groupe de CloudWatch journaux journaux à des fins de débogage et de résolution des problèmes. Les données de journaux peuvent être envoyées à votre groupe de journaux avec ou sans chiffrement AWS KMS à l'aide de votre clé KMS. Pour plus d'informations, consultez Journalisation de la sortie d'actions Automation avec CloudWatch Logs.

- EventBridge Intégration avec Amazon

  L'automatisation est prise en charge en tant que type de cible dans EventBridge les règles Amazon. Cela signifie que vous pouvez déclencher des runbooks à l'aide d'événements. Pour plus d'informations, consultez Surveillance d'événements Systems Manager avec Amazon EventBridge et Référence : modèles et types d'événements Amazon EventBridge pour Systems Manager.

- Partager les bonnes pratiques organisationnelles

  Vous pouvez définir les bonnes pratiques pour la gestion des ressources, les tâches opérationnelles et autres dans les runbooks que vous partagez entre les comptes et les régions.

À qui est destiné Automation ?

- Tout client AWS qui souhaite améliorer son efficacité opérationnelle à grande échelle, réduire les erreurs associées aux interventions manuelles et accélérer la résolution des problèmes courants.
- Experts en infrastructure qui souhaitent automatiser les tâches de déploiement et de configuration.
- Les administrateurs qui souhaitent résoudre de manière fiable les problèmes courants, améliorer l'efficacité du dépannage et réduire les opérations répétitives.
- Les utilisateurs qui souhaitent automatiser une tâche qu'ils exécutent normalement manuellement.

Qu'est-ce qu'une automatisation ?

Une automatisation se compose de toutes les tâches définies dans un runbook et exécutées par le service Automation. Automation utilise les composants suivants pour exécuter des automatisations.

Concept	Détails
Runbook Automation	Un runbook Systems Manager Automation définit l'automatisation (c'est-à-dire les actions exécutées par Systems Manager sur vos nœuds gérés et sur les ressources AWS).

Concept	Détails
	Automation inclut plusieurs runbooks prédéfinis que vous pouvez utiliser afin d'effectuer des tâches courantes, telles que le redémarrage d'une ou plusieurs instances Amazon EC2 ou la création d'une Amazon Machine Image (AMI). Vous pouvez également créer vos propres runbooks. Les runbooks utilisent YAML ou JSON et incluent les étapes et paramètres que vous spécifiez. Les étapes sont exécutées par ordre séquentiel. Pour plus d'informations, consultez Créer vos propres runbooks. Les runbooks sont des documents Systems Manager du type `Automation` , par opposition aux documents `Command`, `Policy` ou `Session`. Les runbooks prennent en charge la version de schéma 0.3. Les documents de commande utilisent la version de schéma 1.2, 2.0, ou 2.2. Les documents de politique utilisent la version de schéma 2.0 ou ultérieure.
Action Automation	L'automatisation définie dans un runbook 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 runbook. Automation prend en charge 20 types d'action différents. Pour plus d'informations, consultez le Référence sur les actions Systems Manager Automation.

Concept	Détails
Quota d'automatisations	Chaque Compte AWS peut exécuter 100 automatisations simultanément. Cela inclut les automatisations enfants (automatisations démarrées par une autre automatisation) et les automatisations de contrôle de débit. Si vous tentez d'en exécuter davantage, Systems Manager ajoute les automatisations supplémentaires à une file d'attente et affiche le statut Pending (En attente). Ce quota peut être ajusté à l'aide de la simultanéité adaptative. Pour plus d'informations, veuillez consulter Permettre à Automation de s'adapter à vos besoins de simultanéité. Pour plus d'informations sur l'exécution d'automatisations, veuillez consulter Exécution d'automatisations.
Quota de mise d'automatisations en file d'attente	Si vous tentez d'exécuter simultanément un nombre d'automatisations supérieur à la limite définie, les automatisations suivantes sont ajoutées à une file d'attente. Chaque Compte AWS peut mettre 5 000 automatisations en file d'attente. Dès qu'une automatisation est terminée (ou a atteint un état terminal), la première automatisation dans la file d'attente commence.

Concept	Détails
Quota d'automatisations de contrôle de débit	Chaque Compte AWS peut exécuter 25 automatisations de contrôle de débit simultanément. Si vous tentez d'exécuter simultanément un nombre d'automatisations de contrôle de débit supérieur à la limite définie, Systems Manager ajoute les automatisations de contrôle de débit suivantes à une file d'attente et affiche le statut « En attente ». Pour de plus amples informations sur l'exécution d'automatisations de contrôle de débit, veuillez consulter Exécution des automatisations à grande échelle.
Quota de file d'attente d'automatisations de contrôle de débit	Si vous tentez d'exécuter simultanément un nombre d'automatisations de contrôle de débit supérieur à la limite définie, les automatisations suivantes sont ajoutées à une file d'attente. Chaque Compte AWS peut mettre 1 000 automatisations de contrôle de débit en file d'attente. Dès qu'une automatisation est terminée (ou a atteint un état terminal), la première automatisation dans la file d'attente commence.

Rubriques

- Configuration d'Automation
- Exécution d'automatisations
- Planification des automatisations
- Référence sur les actions Systems Manager Automation
- Créer vos propres runbooks
- Référence du runbook Systems Manager Automation
- Didacticiels

- Comprendre les statuts d'automatisation
- Résolution des problèmes liés à Systems Manager Automation

Configuration d'Automation

Pour configurer Automation, une fonctionnalité de AWS Systems Manager, vous devez vérifier l'accès des utilisateurs au service Automation et configurer les rôles en fonction de la situation afin que le service puisse effectuer des actions sur vos ressources. Nous vous recommandons également d'activer le mode de simultanéité adaptative dans vos préférences d'Automation. La simultanéité adaptative ajuste automatiquement votre quota d'automatisation pour répondre à vos besoins. Pour plus d'informations, consultez Permettre à Automation de s'adapter à vos besoins de simultanéité.

Pour garantir un accès approprié à AWS Systems Manager Automation, passez en revue les exigences relatives aux rôles d'utilisateur et de service suivantes.

Vérification de l'accès utilisateur aux runbooks

Vérifiez que vous avez l'autorisation d'utiliser des runbooks. Si votre utilisateur, groupe ou rôle 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 politique gérée `AmazonSSMFullAccess` ou une politique dotée d'autorisations de même type à votre utilisateur, groupe ou rôle.

> **Important**
>
> La politique IAM `AmazonSSMFullAccess` octroie des autorisations pour les actions Systems Manager. Toutefois, certains runbooks nécessitent des autorisations pour d'autres services, tels que le runbook `AWS-ReleaseElasticIP`, qui nécessite des autorisations IAM pour `ec2:ReleaseAddress`. Par conséquent, vous devez passer en revue les actions effectuées dans un runbook pour vous assurer que votre utilisateur, groupe ou rôle dispose des autorisations nécessaires pour effectuer les actions incluses dans le runbook.

Configuration d'un accès à un rôle de service (rôle de responsable) pour les automatisations

Les automatisations peuvent être lancées 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'automatisation.

Cependant, les situations suivantes nécessitent que vous spécifiiez un rôle du service pour Automation :

- Lorsque vous souhaitez restreindre les autorisations d'un utilisateur sur une ressource tout en souhaitant que l'utilisateur puisse exécuter une automatisation nécessitant des autorisations supérieurs. Dans ce scénario, vous pouvez créer un rôle de service avec des autorisations supérieures et autoriser l'utilisateur à exécuter l'automatisation.
- Lorsque vous créez une association Systems Manager State Managerqui exécute un runbook.
- Lorsque vous prévoyez que la durée d'exécution de certaines opérations dépasse 12 heures.
- Lorsque vous exécutez un runbook n'appartenant pas à Amazon qui utilise l'`aws:executeScript`action pour appeler une opération d' AWS API ou pour agir sur une AWS ressource. Pour plus d'informations, consultez Autorisations pour l'utilisation de runbooks.

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

Vous pouvez créer un rôle de service pour Automation, une fonctionnalité de AWS Systems Manager, à partir d'un modèle AWS CloudFormation. Après avoir créé le rôle de service, vous pouvez spécifier le rôle de service dans les runbooks à l'aide du paramètre `AutomationAssumeRole`.

Création du rôle de service via AWS CloudFormation

Utilisez la procédure suivante pour créer les rôles AWS Identity and Access Management (IAM) requis pour Systems Manager Automation à l'aide de AWS CloudFormation.

Pour créer le rôle IAM requis

1. Téléchargez et décompressez le fichier AWS-SystemsManager-AutomationServiceRole.zip. Ce fichier 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. Sélectionnez Create Stack (Créer une pile).
4. Dans la section Spécifier un modèle, sélectionnez Charger un modèle de fichier.
5. Sélectionnez Browse (Parcourir), puis sélectionnez le fichier modèle AWS CloudFormation `AWS-SystemsManager-AutomationServiceRole.yaml`.
6. Sélectionnez 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. Sélectionnez Suivant.
9. Faites défiler la page Vérification vers le bas et sélectionnez l'option Je comprends qu'AWS CloudFormation peut créer des ressources IAM.
10. Sélectionnez Create (Créer).

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 runbooks Automation d'AWS (runbooks `AWS-*`) tels que les runbooks `AWS-ConfigureS3BucketLogging`, `AWS-CreateDynamoDBBackup` et `AWS-`

`RestartEC2Instance`, par exemple. Cette exigence s'applique également à tous les runbooks 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 octroyer des autorisations à d'autres Services AWS en ajoutant une politique IAM en ligne au rôle. Pour de plus amples informations, veuillez consulter (Facultatif) Ajoutez une politique d'automatisation en ligne ou une politique gérée par le client pour invoquer d'autres Services AWS.

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 utilisez un runbook.

> **Note**
>
> Vous n'avez pas besoin de copier les informations du rôle en utilisant cette procédure si vous exécutez les runbooks `AWS-UpdateLinuxAmi` ou `AWS-UpdateWindowsAmi`. Ces runbooks possèdent déjà les rôles requis spécifiés comme valeurs par défaut. Ces rôles spécifiés dans ces runbooks utilisant des politiques 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. Sélectionnez l'onglet Ressources.
4. Sélectionnez le lien ID physique pour AutomationServiceRole. La console IAM ouvre un récapitulatif du rôle de service Automation.
5. Copiez l'Amazon Resource Name (ARN) 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 runbooks.

Méthode 2 : Utiliser IAM afin de configurer des rôles pour Automation

Si vous devez créer un rôle de service pour Automation, une fonctionnalité de AWS Systems Manager, effectuez les tâches suivantes. Pour plus d'informations sur le moment où un rôle de service est requis pour Automation, consultez Configuration d'Automation.

Tâches

- Tâche 1 : Création d'un rôle de service pour Automation
- Tâche 2 : associer la PassRole politique iam : à votre rôle d'automatisation

Tâche 1 : Création d'un rôle de service pour Automation

Utilisez la procédure suivante pour créer un rôle de service (ou endosser un rôle) pour Systems Manager Automation.

> **Note**
>
> Vous pouvez également utiliser ce rôle dans des runbooks, notamment le runbook `AWS-CreateManagedLinuxInstance`. L'utilisation de ce rôle, ou de l'Amazon Resource Name (ARN) d'un rôle AWS Identity and Access Management (IAM), dans les runbooks permet à Automation d'effectuer des actions dans votre environnement, telles que le lancement de nouvelles instances et des actions en votre nom.

Pour créer un rôle IAM et autoriser Automation à l'endosser

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le volet de navigation, sélectionnez Rôles, puis Créer un rôle.
3. Sous Select type of trusted entity, sélectionnez AWS service.
4. Dans la section Choisir un cas d'utilisation sélectionnez Systems Manager, puis sélectionnez Suivant : Autorisations.
5. Sur la page Politique d'autorisation jointe, recherchez la AutomationRole politique AmazonSSM, choisissez-la, puis choisissez Suivant : Réviser.

6. Sur la page Review (Vérification), saisissez un nom dans la zone Role name (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, sélectionnez 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 associerez la PassRole politique iam : à votre compte IAM dans la procédure suivante. Vous pouvez également spécifier le nom du rôle et l'ARN dans des runbooks.

> **Note**
>
> La `AmazonSSMAutomationRole` politique attribue l'autorisation du rôle Automation à un sous-ensemble de AWS Lambda fonctions 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 des fonctions Lambda existantes dont les ARN n'utilisent pas ce format, vous devez également associer une politique Lambda supplémentaire à votre rôle d'automatisation, telle que la politique. AWSLambdaRole La politique ou le rôle supplémentaire doit fournir un accès plus large aux fonctions Lambda au sein du Compte AWS.

Après avoir créé votre fonction du service, nous vous recommandons de modifier la politique d'approbation afin d'éviter le problème de député confus entre services. Le problème de député confus est un problème de sécurité dans lequel une entité qui n'est pas autorisée à effectuer une action peut contraindre une entité plus privilégiée à le faire. En AWS, l'usurpation d'identité interservices peut entraîner la confusion des adjoints. L'usurpation d'identité entre services peut se produire lorsqu'un service (le service appelant) appelle un autre service (le service appelé). Le service appelant peut être manipulé et ses autorisations utilisées pour agir sur les ressources d'un autre client auxquelles on ne serait pas autorisé d'accéder autrement. Pour éviter cela, AWS fournit des outils qui vous aident à protéger vos données pour tous les services auprès des principaux fournisseurs de services qui ont obtenu l'accès aux ressources de votre compte.

Nous vous recommandons d'utiliser les clés de contexte de condition globale `aws:SourceArn` et `aws:SourceAccount` dans les politiques de ressources pour limiter les autorisations à la ressource octroyées par Automation à un autre service. Si la valeur `aws:SourceArn` ne contient pas l'ID de

compte, tel qu'un ARN de compartiment Amazon S3, vous devez utiliser les deux clés de contexte de condition globale pour limiter les autorisations. Si vous utilisez les deux clés de contexte de condition globale et que la valeur `aws:SourceArn` contient l'ID de compte, la valeur `aws:SourceAccount` et le compte dans la valeur `aws:SourceArn` doivent utiliser le même ID de compte lorsqu'ils sont utilisés dans la même instruction de politique. Utilisez `aws:SourceArn` si vous souhaitez qu'une seule ressource soit associée à l'accès entre services. Utilisez `aws:SourceAccount` si vous souhaitez autoriser l'association d'une ressource de ce compte à l'utilisation interservices. La valeur de `aws:SourceArn` doit être l'ARN pour les exécutions d'automatisation. Si vous ne connaissez pas l'ARN complet de la ressource ou si vous spécifiez plusieurs ressources, utilisez la clé de contexte de condition globale `aws:SourceArn` avec des caractères génériques (*) pour les parties inconnues de l'ARN. Par exemple, `arn:aws:ssm:*:123456789012:automation-execution/*`.

L'exemple suivant montre comment utiliser les clés de contexte de condition globale `aws:SourceArn` et `aws:SourceAccount` pour Automation afin d'éviter le problème du député confus.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "ssm.amazonaws.com"
        ]
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "123456789012"
        },
        "ArnLike": {
          "aws:SourceArn": "arn:aws:ssm:*:123456789012:automation-execution/*"
        }
      }
    }
  ]
}
```

Pour modifier la politique d'approbation du rôle

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation, sélectionnez Rôles.
3. Dans la liste des rôles de votre compte, sélectionnez le nom de votre rôle de service Automation.
4. Sélectionnez l'onglet Relations d'approbation, puis Modifier la relation d'approbation.
5. Modifiez la politique d'approbation à l'aide des clés de contexte de condition globale `aws:SourceArn` et `aws:SourceAccount` pour Automation afin d'éviter tout problème de député confus.
6. Pour enregistrer vos modifications, sélectionnez Update Trust Policy (Mettre à jour la politique d'approbation).

(Facultatif) Ajoutez une politique d'automatisation en ligne ou une politique gérée par le client pour invoquer d'autres Services AWS

Si vous exécutez une automatisation qui Services AWS en appelle d'autres 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 runbooks AWS Automation (`AWS-*`runbooks) tels que, et `AWS-RestartEC2Instance` runbooks `AWS-ConfigureS3BucketLoggingAWS-CreateDynamoDBBackup`, pour n'en nommer que quelques-uns. Cette exigence s'applique également à tous les runbooks 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 accorder des autorisations à d'autres Services AWS personnes en ajoutant au rôle une politique en ligne IAM ou une politique gérée par le client.

Pour intégrer une politique en ligne à un rôle de service (console IAM)

1. Connectez-vous à la console IAM AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation, sélectionnez Rôles.
3. Dans la liste, sélectionnez le nom du rôle que vous souhaitez modifier.
4. Sélectionnez l'onglet Autorisations.
5. Dans le menu déroulant Ajouter des autorisations, choisissez Joindre des politiques ou Créer une politique en ligne.

6. Si vous choisissez Joindre des politiques, cochez la case située à côté de la politique que vous souhaitez ajouter et choisissez Ajouter des autorisations.
7. Si vous choisissez Créer une politique en ligne, sélectionnez l'onglet JSON.
8. Entrez un document de politique JSON pour le que Services AWS vous souhaitez invoquer. Voici deux exemples de document de politique 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 et exemple CreateSnapshot DescribeSnapShots

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":"ec2:CreateSnapshot",
         "Resource":"*"
      },
      {
         "Effect":"Allow",
         "Action":"ec2:DescribeSnapshots",
         "Resource":"*"
      }
   ]
}
```

Pour obtenir des détails sur la terminologie IAM, veuillez consulter la Référence de politique JSON IAM dans le Guide de l'utilisateur IAM.

9. Lorsque vous avez terminé, sélectionnez Review policy (Examiner une politique). Le programme de validation de politique signale les éventuelles erreurs de syntaxe.
10. Sur la page Review policy (Examiner une politique), saisissez un Name (Nom) pour la politique que vous êtes en train de créer. Vérifiez le récapitulatif de politique pour voir les autorisations accordées par votre politique. Sélectionnez ensuite Créer une politique pour enregistrer votre travail.
11. Une fois que vous avez créé une politique en ligne, elle est automatiquement intégrée à votre rôle.

Tâche 2 : associer la PassRole politique iam : à votre rôle d'automatisation

Utilisez la procédure suivante pour attacher la politique `iam:PassRole` à votre rôle de service Automation. Cela permet au service Automation de transmettre le rôle à d'autres services ou fonctionnalités de Systems Manager lors de l'exécution d'automatisations.

Pour associer la PassRole politique iam : à votre rôle d'automatisation

1. Dans la page Récapitulatif du rôle que vous venez de créer, sélectionnez l'onglet Autorisations.
2. Sélectionnez Ajouter une politique en ligne.
3. Dans la page Créer une politique, sélectionnez l'onglet Éditeur visuel.
4. Sélectionnez Service, puis sélectionnez IAM.
5. Sélectionnez Sélectionner des actions.
6. Dans la zone de texte Actions de filtrage**`PassRole`**, tapez, puis choisissez l'PassRoleoption.
7. Sélectionnez Ressources. Vérifiez que Spécifique est sélectionné, puis sélectionnez 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 Ajouter.
10. Sélectionnez Review policy (Examiner une politique).
11. Sur la page Review Policy (Examiner une politique), saisissez un nom, puis sélectionnez Create Policy (Créer une politique).

Permettre à Automation de s'adapter à vos besoins de simultanéité

Par défaut, Automation vous permet d'exécuter jusqu'à 100 automatisations simultanées à la fois. Automation fournit également un paramètre facultatif que vous pouvez utiliser pour ajuster automatiquement votre quota d'automatisation de la simultanéité. Avec ce paramètre, votre quota d'automatisation de la simultanéité peut prendre en charge jusqu'à 500 automatisations simultanées, en fonction des ressources disponibles.

> **Note**
>
> Si votre automatisation appelle des opérations d'API, une mise à l'échelle adaptative à vos cibles peut entraîner des exceptions de limitation. Si des exceptions de limitation récurrentes se produisent lors de l'exécution d'automatisations alors que la simultanéité adaptative est activée, vous devrez peut-être demander des augmentations de quota pour l'opération d'API, si possible.

Pour activer la concurrence adaptative (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation de gauche, sélectionnez Automation (Automatisation).
3. Sélectionnez l'onglet Préférences, puis Modifier.
4. Cochez la case en regard de l'option Enable adaptive concurrency (Activer la simultanéité adaptative).
5. Choisissez Enregistrer.

Mise en place de restrictions des modifications dans Automation

Par défaut, Automation vous permet d'utiliser des runbooks sans contraintes de date ni d'heure. En intégrant Automation à Change Calendar, vous pouvez implémenter des restrictions de modification pour toutes les automatisations de votre Compte AWS. Avec ce paramètre, les principaux AWS Identity and Access Management (IAM) de votre compte ne peuvent exécuter des automatisations que durant les périodes autorisées par votre calendrier des modifications. Pour plus d'informations sur l'utilisation avec Change Calendar, consultez Utilisation des Change Calendar.

Pour activer les commandes de modification (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 de gauche, sélectionnez Automation (Automatisation).
3. Sélectionnez l'onglet Préférences, puis Modifier.
4. Cochez la case située à côté de l'option Activer l'intégration de Change Calendar.
5. Dans la liste déroulante Choisir un calendrier des modifications, choisissez le calendrier des modifications que vous souhaitez qu'Automation suive.
6. Choisissez Enregistrer.

Exécution d'automatisations

Cette section comprend des informations sur le mode d'exécution des runbooks Automation. Automation est une fonctionnalité de AWS Systems Manager. Pour des didacticiels plus détaillés sur la façon d'exécuter des automatisations pour votre cas d'utilisation, veuillez consulter la rubrique Didacticiels.

Table des matières

Exécution de l'automatisation.

Lorsque vous exécutez une automatisation, celle-ci est exécutée par défaut dans le cas de l'utilisateur qui a lancé l'exécution. Cela signifie, par exemple, que si votre utilisateur dispose d'autorisations d'administration, l'automatisation est exécutée avec les autorisations d'administrateur et un accès total aux ressources configurées par l'automatisation. La bonne pratique en matière de sécurité consiste à exécuter les automatisations à l'aide d'un rôle de service IAM également considéré dans ce cas comme rôle endossé configuré avec la politique de gestion AmazonSSMAutomationRole. Afin d'utiliser différents runbooks, vous devrez peut-être ajouter des politiques IAM supplémentaires à votre rôle d'utilisateur. L'utilisation d'un rôle de service IAM pour exécuter une automatisation est nommée administration déléguée.

Lorsque vous utilisez un rôle de service, l'automatisation peut être exécuté sur les ressources AWS, mais l'utilisateur qui a exécuté l'automatisation dispose d'un accès restreint (ou nul) à ces ressources. Par exemple, vous pouvez configurer un rôle de service et l'utiliser avec Automation afin de redémarrer une ou plusieurs instances Amazon Elastic Compute Cloud (Amazon EC2). Automation est une fonctionnalité de AWS Systems Manager. 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 lorsque vous exécutez une automatisation, ou vous pouvez créer des runbooks personnalisés et spécifier le rôle de service directement dans le runbook. Si vous spécifiez un rôle de service au moment de l'exécution ou dans un runbook, le service est alors exécuté dans le contexte du rôle de service spécifié. Si vous ne spécifiez aucun 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 qui sera probablement exécutée pendant plus de 12 heures. Si vous lancez une exécution d'automatisation de longue durée dans le contexte d'un utilisateur, la session temporaire de ce dernier expire après 12 heures.

L'administration déléguée garantit un niveau de sécurité élevé et le contrôle de vos ressources AWS. Il 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.

Avant de commencer

Avant d'exécuter les procédures suivantes, vous devez créer la fonction de service IAM et configurer une relation d'approbation pour la fonctionnalité Automation d'AWS Systems Manager. Pour de plus amples informations, veuillez consulter Tâche 1 : Création d'un rôle de service pour Automation.

La procédure suivante décrit la méthode d'utilisation de la console Systems Manager afin d'exécuter une simple automatisation.

Exécution d'une automatisation simple (console)

La procédure suivante décrit comment utiliser la console Systems Manager pour exécuter une simple automatisation.

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, sélectionnez Automation (Automatisation), puis Execute automation (Exécuter l'automatisation).
3. Dans la liste Automation document (Document Automation), sélectionnez un runbook. Sélectionnez une ou plusieurs options dans le panneau Document categories (Catégories de documents) pour filtrer les documents SSM en fonction de leur but. Pour afficher un runbook vous appartenant, sélectionnez l'onglet Owned by me (M'appartenant). Pour afficher un runbook partagé avec votre compte, sélectionnez l'onglet Shared with me (Partagé avec moi). Pour afficher tous les runbooks, sélectionnez l'onglet All documents (Tous les documents).

   > **Note**
   >
   > Vous pouvez consulter les informations sur un runbook en sélectionnant 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 :

   - Version par défaut lors de l'exécution : sélectionnez cette option si le runbook d'automatisation est mis à jour régulièrement et qu'une nouvelle version par défaut est attribuée.
   - Dernière version lors de l'exécution : sélectionnez cette option si le runbook d'automatisation est mis à jour régulièrement et que vous souhaitez exécuter la dernière version mise à jour.
   - 1 (Par défaut) : sélectionnez cette option pour exécuter la première version du document, qui est la version par défaut.

5. Choisissez Next (Suivant).
6. Dans la section Execution Mode (Mode d'exécution), sélectionnez 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. (Facultatif) Choisissez une alarme CloudWatch à appliquer à votre automatisation à des fins de surveillance. Afin d'associer une alarme CloudWatch à votre automatisation, le principal IAM démarrant l'automatisation doit disposer de l'autorisation pour l'action `iam:createServiceLinkedRole`. Pour de plus amples informations relatives à la configuration des alarmes CloudWatch, consultez Utilisation des alarmes Amazon CloudWatch. Notez que l'activation de votre alarme arrête l'automatisation. Si vous utilisez AWS CloudTrail, vous verrez l'appel d'API dans votre journal.
9. Sélectionnez Execute (Exécuter).

La console affiche le statut de l'automatisation. Si l'exécution de l'automatisation échoue, consultez Résolution des problèmes liés à Systems Manager Automation.

Exécution d'une automatisation simple (ligne de commande)

La procédure suivante décrit comment utiliser l'AWS CLI (sous Linux ou Windows) ou AWS Tools for PowerShell pour exécuter une automatisation simple.

Pour exécuter une automatisation simple

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

   Pour plus d'informations, consultez la section Installation ou mise à jour de la version la plus récente de l'AWS CLI et Installation d'AWS Tools for PowerShell.
2. Exécutez la commande suivante pour démarrer une automatisation simple. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations.

   Linux & macOS

   ```
   aws ssm start-automation-execution \
       --document-name runbook name \
       --parameters runbook parameters
   ```

Windows

```
aws ssm start-automation-execution ^
    --document-name runbook name ^
    --parameters runbook parameters
```

PowerShell

```
Start-SSMAutomationExecution `
  -DocumentName runbook name `
  -Parameter runbook parameters
```

Voici un exemple d'utilisation du runbook `AWS-RestartEC2Instance` pour redémarrer l'instance EC2 spécifiée.

Linux & macOS

```
aws ssm start-automation-execution \
    --document-name "AWS-RestartEC2Instance" \
    --parameters "InstanceId=i-02573cafcfEXAMPLE"
```

Windows

```
aws ssm start-automation-execution ^
    --document-name "AWS-RestartEC2Instance" ^
    --parameters "InstanceId=i-02573cafcfEXAMPLE"
```

PowerShell

```
Start-SSMAutomationExecution `
  -DocumentName AWS-RestartEC2Instance `
  -Parameter @{"InstanceId"="i-02573cafcfEXAMPLE"}
```

Le système retourne des informations telles que les suivantes.

Linux & macOS

```
{
```

```
    "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 & macOS

   ```
   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 retourne des informations telles que les suivantes.

   Linux & macOS

   ```
   {
       "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 avec des approbateurs

Les procédures suivantes décrivent comment utiliser la console AWS Systems Manager et l'AWS Command Line Interface (AWS CLI) pour exécuter une automatisation avec des approbations à l'aide d'une exécution simple. L'automatisation utilise l'action Automation `aws:approve`, qui interrompt temporairement l'automatisation jusqu'à ce que les principaux désignés approuvent ou refusent l'action. Le flux de travail Automation s'exécute dans le contexte de l'utilisateur actuel. Cela signifie que vous n'avez pas besoin de configurer d'autorisations IAM supplémentaires tant que vous avez le droit d'utiliser le runbook et les actions qu'il appelle. Si vous disposez des autorisations d'administrateur dans IAM, vous pouvez exécuter ce runbook.

Avant de commencer

En plus des entrées standard requises par le runbook, 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 nom d'utilisateur ou d'un ARN d'utilisateur. Si plusieurs approbateurs sont fournis, un nombre minimum d'approbations correspondant doit également être spécifié dans le runbook.
- 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 plus d'informations, consultez Créer une rubrique dans le Guide du développeur d'Amazon Simple Notification Service.

Exécution d'une automatisation avec des approbateurs (console)

Pour exécuter une automatisation avec des approbateurs

La procédure suivante décrit comment utiliser la console Systems Manager pour exécuter une simple 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, sélectionnez Automation (Automatisation), puis Execute automation (Exécuter l'automatisation).
3. Dans la liste Automation document (Document Automation), sélectionnez un runbook. Sélectionnez une ou plusieurs options dans le panneau Document categories (Catégories de documents) pour filtrer les documents SSM en fonction de leur but. Pour afficher un runbook vous appartenant, sélectionnez l'onglet Owned by me (M'appartenant). Pour afficher un runbook partagé avec votre compte, sélectionnez l'onglet Shared with me (Partagé avec moi). Pour afficher tous les runbooks, sélectionnez l'onglet All documents (Tous les documents).

   > **Note**
   >
   > Vous pouvez consulter les informations sur un runbook en sélectionnant 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 :

   - Version par défaut lors de l'exécution : sélectionnez cette option si le runbook d'automatisation est mis à jour régulièrement et qu'une nouvelle version par défaut est attribuée.
   - Dernière version lors de l'exécution : sélectionnez cette option si le runbook d'automatisation est mis à jour régulièrement et que vous souhaitez exécuter la dernière version mise à jour.
   - 1 (Par défaut) : sélectionnez cette option pour exécuter la première version du document, qui est la version par défaut.
5. Choisissez Next (Suivant).
6. Sur la page Execute automation document (Exécuter le document Automation), sélectionnez Simple execution (Exécution simple).
7. Dans la section Paramètres d'entrée, spécifiez les paramètres d'entrée obligatoires.

   Par exemple, si vous avez choisi le runbook **`AWS-StartEC2InstanceWithApproval`**, vous devez spécifier ou choisir des ID d'instance pour le paramètre InstanceId.
8. Dans la section Approbateurs, spécifiez les noms d'utilisateurs ou les ARN d'utilisateur des approbateurs pour l'action Automation.
9. Dans la section SNSTopicARN, spécifiez l'ARN de la rubrique SNS à utiliser pour l'envoi de notifications d'approbation. Le nom de la rubrique SNS doit commencer par Automation.
10. Le cas échéant, vous pouvez choisir un rôle de service IAM dans la liste AutomationAssumeRole. Si vous ciblez plus de 100 comptes et régions, vous devez spécifier le `AWS-SystemsManager-AutomationAdministrationRole`.
11. Sélectionnez Execute automation (Exécuter l'automatisation).

L'approbateur spécifié reçoit une notification Amazon SNS 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 choisissez d'approuver l'automatisation, l'automatisation continue d'exécuter les étapes incluses dans le runbook spécifié. La console affiche le statut de l'automatisation. Si l'exécution de l'automatisation échoue, consultez Résolution des problèmes liés à Systems Manager Automation.

Pour approuver ou rejeter 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, sélectionnez Automation, puis sélectionnez l'automatisation qui a été exécuté au cours de la procédure précédente.
3. Sélectionnez Actions puis Approve/Deny (Approuver/Refuser).
4. Sélectionnez d'Approuver ou de Refuser et, le cas échéant, saisissez un commentaire.
5. Sélectionnez Submit (Envoyer).

Exécution d'une automatisation avec des approbateurs (ligne de commande)

La procédure suivante décrit comment utiliser l'AWS CLI (sous Linux ou Windows) ou les AWS Tools for PowerShell pour exécuter une automatisation avec des approbateurs.

Pour exécuter une automatisation avec des approbateurs

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

   Pour plus d'informations, consultez la section Installation ou mise à jour de la version la plus récente de l'AWS CLI et Installation d'AWS Tools for PowerShell.
2. Exécutez la commande suivante pour lancer une automatisation avec des approbateurs. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations. Dans la section Document name (Nom du document), spécifiez un runbook qui inclut l'action Automation, `aws:approve`.

   Pour `Approvers`, spécifiez les noms d'utilisateurs ou les ARN d'utilisateur des approbateurs de l'action. Pour `SNSTopic`, spécifiez l'ARN de la rubrique SNS à utiliser pour envoyer la notification d'approbation. Le nom de la rubrique Amazon 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 runbook que vous sélectionnez.

Linux & macOS

```
aws ssm start-automation-execution \
    --document-name "AWS-StartEC2InstanceWithApproval" \
    --parameters
 "InstanceId=i-02573cafcfEXAMPLE,Approvers=arn:aws:iam::123456789012:role/
Administrator,SNSTopicArn=arn:aws:sns:region:123456789012:AutomationApproval"
```

Windows

```
aws ssm start-automation-execution ^
    --document-name "AWS-StartEC2InstanceWithApproval" ^
    --parameters
 "InstanceId=i-02573cafcfEXAMPLE,Approvers=arn:aws:iam::123456789012:role/
Administrator,SNSTopicArn=arn:aws:sns:region:123456789012:AutomationApproval"
```

PowerShell

```
Start-SSMAutomationExecution `
    -DocumentName AWS-StartEC2InstanceWithApproval `
    -Parameters @{
        "InstanceId"="i-02573cafcfEXAMPLE"
        "Approvers"="arn:aws:iam::123456789012:role/Administrator"
        "SNSTopicArn"="arn:aws:sns:region:123456789012:AutomationApproval"
    }
```

Le système retourne des informations telles que les suivantes.

Linux & macOS

```
{
    "AutomationExecutionId": "df325c6d-b1b1-4aa0-8003-6cb7338213c6"
}
```

Windows

```
{
    "AutomationExecutionId": "df325c6d-b1b1-4aa0-8003-6cb7338213c6"
```

```
}
```

PowerShell

```
df325c6d-b1b1-4aa0-8003-6cb7338213c6
```

Pour approuver une automatisation

- Exécutez la commande suivante pour approuver une automatisation. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

  Linux & macOS

  ```
  aws ssm send-automation-signal \
      --automation-execution-id "df325c6d-b1b1-4aa0-8003-6cb7338213c6" \
      --signal-type "Approve" \
      --payload "Comment=your comments"
  ```

  Windows

  ```
  aws ssm send-automation-signal ^
      --automation-execution-id "df325c6d-b1b1-4aa0-8003-6cb7338213c6" ^
      --signal-type "Approve" ^
      --payload "Comment=your comments"
  ```

  PowerShell

  ```
  Send-SSMAutomationSignal `
      -AutomationExecutionId df325c6d-b1b1-4aa0-8003-6cb7338213c6 `
      -SignalType Approve `
      -Payload @{"Comment"="your comments"}
  ```

  Il n'y a pas de sortie si la commande réussit.

Pour refuser une automatisation

- Exécutez la commande suivante pour refuser une automatisation. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

  Linux & macOS

  ```
  aws ssm send-automation-signal \
      --automation-execution-id "df325c6d-b1b1-4aa0-8003-6cb7338213c6" \
      --signal-type "Deny" \
      --payload "Comment=your comments"
  ```

  Windows

  ```
  aws ssm send-automation-signal ^
      --automation-execution-id "df325c6d-b1b1-4aa0-8003-6cb7338213c6" ^
      --signal-type "Deny" ^
      --payload "Comment=your comments"
  ```

  PowerShell

  ```
  Send-SSMAutomationSignal `
      -AutomationExecutionId df325c6d-b1b1-4aa0-8003-6cb7338213c6 `
      -SignalType Deny `
      -Payload @{"Comment"="your comments"}
  ```

  Il n'y a pas de sortie si la commande réussit.

Exécution des automatisations à grande échelle

Avec AWS Systems Manager Automatisation, il est possible d'exécuter des automatisations sur une flotte AWS ressources en utilisant des cibles. De plus, vous pouvez contrôler le déploiement de l'automatisation dans l'ensemble de la flotte en spécifiant une valeur de simultanéité et un seuil d'erreurs. Les fonctions de simultanéité et de seuil d'erreurs sont appelées collectivement contrôles du débit. La valeur de simultanéité détermine le nombre de ressources autorisées à exécuter l'automatisation simultanément. Automation fournit également un mode de simultanéité adaptative auquel vous pouvez vous inscrire. La simultanéité adaptative met automatiquement à l'échelle votre quota d'automatisation en passant de 100 automatisations exécutées simultanément à 500. Le

seuil d'erreurs détermine le nombre d'automatisations qui peuvent échouer jusqu'à ce que Systems Manager cesse d'envoyer l'automatisation à d'autres ressources.

Pour de plus amples informations sur la simultanéité et les seuils d'erreurs, consultez Automatisations de contrôle à grande échelle. Pour de plus amples informations sur les cibles, consultez Mappage des cibles pour une automatisation.

Les procédures suivantes décrivent comment activer la simultanéité adaptative et exécuter une automatisation avec des cibles et des contrôles de taux à l'aide de la console Systems Manager et l'AWS Command Line Interface (AWS CLI).

Exécution d'une automatisation avec des cibles et des contrôles du débit (console)

La procédure suivante décrit comment utiliser la console Systems Manager 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, sélectionnez Automation (Automatisation), puis Execute automation (Exécuter l'automatisation).
3. Dans la liste Automation document (Document Automation), sélectionnez un runbook. Sélectionnez une ou plusieurs options dans le panneau Document categories (Catégories de documents) pour filtrer les documents SSM en fonction de leur but. Pour afficher un runbook vous appartenant, sélectionnez l'onglet Owned by me (M'appartenant). Pour afficher un runbook partagé avec votre compte, sélectionnez l'onglet Shared with me (Partagé avec moi). Pour afficher tous les runbooks, sélectionnez l'onglet All documents (Tous les documents).

   > **Note**
   >
   > Vous pouvez consulter les informations sur un runbook en sélectionnant 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 :

   - Version par défaut lors de l'exécution : sélectionnez cette option si le runbook d'automatisation est mis à jour régulièrement et qu'une nouvelle version par défaut est attribuée.

- Dernière version lors de l'exécution : sélectionnez cette option si le runbook d'automatisation est mis à jour régulièrement et que vous souhaitez exécuter la dernière version mise à jour.
- 1 (Par défaut) : sélectionnez cette option pour exécuter la première version du document, qui est la version par défaut.

5. Choisissez Next (Suivant).
6. Dans la section Execution Mode (Mode d'exécution), sélectionnez 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), sélectionnez 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 runbook Automation 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 choisissez de cibler des ressources à l'aide de valeurs de paramètre, saisissez la valeur de paramètre du paramètre que vous avez choisi, dans la section Input parameters (Paramètres d'entrée).

      ii. Si vous choisissez de cibler les ressources avec AWS Resource Groups, sélectionnez 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 runbook Automation sur toutes les instances du Compte AWS et de la Région AWS actuels, 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 pourriez ne pas avoir besoin de choisir certaines options dans la section 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 runbook `AWS-`

`RestartEC2Instance`, vous n'avez pas à spécifier ou à choisir d'ID d'instance dans la section Inpunt parameters (Paramètres d'entrée). L'exécution d'Automation 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 Simultanéité, sélectionnez une option :

   - Sélectionnez targets (cibles) pour entrer un nombre absolu de cibles pouvant exécuter le flux de travail Automation simultanément.
   - Sélectionnez 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), sélectionnez une option :

    - Sélectionnez erreurs pour indiquer un nombre absolu d'erreurs autorisées avant qu'Automation ne cesse d'envoyer le flux de travail à d'autres ressources.
    - Sélectionnez pourcentage pour indiquer un pourcentage d'erreurs autorisées avant qu'Automation ne cesse d'envoyer le flux de travail à d'autres ressources.

11. (Facultatif) Choisissez une alarme CloudWatch à appliquer à votre automatisation à des fins de surveillance. Afin d'associer une alarme CloudWatch à votre automatisation, le principal IAM démarrant l'automatisation doit disposer de l'autorisation pour l'action `iam:createServiceLinkedRole`. Pour de plus amples informations relatives à la configuration des alarmes CloudWatch, consultez Utilisation des alarmes Amazon CloudWatch. Notez que l'activation de votre alarme arrête l'automatisation. Si vous utilisez AWS CloudTrail, vous verrez l'appel d'API dans votre journal.

12. Sélectionnez Execute (Exécuter).

Pour afficher les automatisations démarrées par votre automatisation du contrôle de débit, dans le panneau de navigation, sélectionnez Automation (Automatisation), puis 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 les AWS Tools for 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 AWS Tools for PowerShell si vous ne l'avez pas déjà fait.

   Pour plus d'informations, consultez la section Installation ou mise à jour de la version la plus récente de l'AWS CLI et Installation d'AWS Tools for PowerShell.

2. Exécutez la commande suivante pour afficher une liste de documents.

   Linux & macOS

   ```
   aws ssm list-documents
   ```

   Windows

   ```
   aws ssm list-documents
   ```

   PowerShell

   ```
   Get-SSMDocumentList
   ```

   Notez le nom du runbook à utiliser.

3. Exécutez la commande suivante pour afficher les détails relatifs au runbook. Remplacez *runbook name* (nom du runbook) par le nom du runbook pour lequel vous souhaitez afficher les détails. 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 & macOS

   ```
   aws ssm describe-document \
       --name runbook name
   ```

   Windows

   ```
   aws ssm describe-document ^
       --name runbook name
   ```

PowerShell

```
Get-SSMDocumentDescription `
    -Name runbook name
```

4. Créez une commande qui utilise les options de cibles et de contrôle du débit que vous souhaitez exécuter. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   Ciblage à l'aide de balises

   Linux & macOS

```
aws ssm start-automation-execution \
    --document-name runbook name \
    --targets Key=tag:key name,Values=value \
    --target-parameter-name parameter name \
    --parameters "input parameter name=input parameter value,input parameter 2
 name=input parameter 2 value" \
    --max-concurrency 10 \
    --max-errors 25%
```

   Windows

```
aws ssm start-automation-execution ^
    --document-name runbook name ^
    --targets Key=tag:key name,Values=value ^
    --target-parameter-name parameter name ^
    --parameters "input parameter name=input parameter value,input parameter 2
 name=input parameter 2 value" ^
    --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 "runbook name" `
```

```
    -Targets $Targets `
    -TargetParameterName "parameter name" `
    -Parameter @{"input parameter name"="input parameter value";"input parameter
 2 name"="input parameter 2 value"} `
    -MaxConcurrency "10" `
    -MaxError "25%"
```

Ciblage à l'aide des valeurs de paramètre

Linux & macOS

```
aws ssm start-automation-execution \
    --document-name runbook name \
    --targets Key=ParameterValues,Values=value,value 2,value 3 \
    --target-parameter-name parameter name \
    --parameters "input parameter name=input parameter value" \
    --max-concurrency 10 \
    --max-errors 25%
```

Windows

```
aws ssm start-automation-execution ^
    --document-name runbook name ^
    --targets Key=ParameterValues,Values=value,value 2,value 3 ^
    --target-parameter-name parameter name ^
    --parameters "input parameter name=input parameter value" ^
    --max-concurrency 10 ^
    --max-errors 25%
```

PowerShell

```
$Targets = New-Object Amazon.SimpleSystemsManagement.Model.Target
$Targets.Key = "ParameterValues"
$Targets.Values = "value","value 2","value 3"

Start-SSMAutomationExecution `
    -DocumentName "runbook name" `
    -Targets $Targets `
    -TargetParameterName "parameter name" `
    -Parameter @{"input parameter name"="input parameter value"} `
    -MaxConcurrency "10" `
```

```
    -MaxError "25%"
```

Ciblage à l'aide d'AWS Resource Groups

Linux & macOS

```
aws ssm start-automation-execution \
    --document-name runbook name \
    --targets Key=ResourceGroup,Values=Resource group nname \
    --target-parameter-name parameter name \
    --parameters "input parameter name=input parameter value" \
    --max-concurrency 10 \
    --max-errors 25%
```

Windows

```
aws ssm start-automation-execution ^
    --document-name runbook name ^
    --targets Key=ResourceGroup,Values=Resource group name ^
    --target-parameter-name parameter name ^
    --parameters "input parameter name=input parameter value" ^
    --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 "runbook name" `
    -Targets $Targets `
    -TargetParameterName "parameter name" `
    -Parameter @{"input parameter name"="input parameter value"} `
    -MaxConcurrency "10" `
    -MaxError "25%"
```

Ciblage de toutes les instances Amazon EC2 de l'Compte AWS et de l'Région AWS actuels

Linux & macOS

```
aws ssm start-automation-execution \
    --document-name runbook name \
    --targets "Key=AWS::EC2::Instance,Values=*"  \
    --target-parameter-name instanceId \
    --parameters "input parameter name=input parameter value" \
    --max-concurrency 10 \
    --max-errors 25%
```

Windows

```
aws ssm start-automation-execution ^
    --document-name runbook name ^
    --targets Key=AWS::EC2::Instance,Values=* ^
    --target-parameter-name instanceId ^
    --parameters "input parameter name=input parameter value" ^
    --max-concurrency 10 ^
    --max-errors 25%
```

PowerShell

```
$Targets = New-Object Amazon.SimpleSystemsManagement.Model.Target
$Targets.Key = "AWS::EC2::Instance"
$Targets.Values = "*"

Start-SSMAutomationExecution `
    -DocumentName "runbook name" `
    -Targets $Targets `
    -TargetParameterName "instanceId" `
    -Parameter @{"input parameter name"="input parameter value"} `
    -MaxConcurrency "10" `
    -MaxError "25%"
```

La commande renvoie un ID d'exécution. Copiez cet ID dans le Presse-papiers. Vous pouvez utiliser cet ID pour afficher l'état de l'automatisation.

Linux & macOS

```
{
```

```
    "AutomationExecutionId": "a4a3c0e9-7efd-462a-8594-01234EXAMPLE"
}
```

Windows

```
{
    "AutomationExecutionId": "a4a3c0e9-7efd-462a-8594-01234EXAMPLE"
}
```

PowerShell

```
a4a3c0e9-7efd-462a-8594-01234EXAMPLE
```

5. Pour afficher l'automatisation, exécutez la commande suivante. Remplacez chaque *automation execution ID* (ID d'exécution de l'automatisation) par vos propres informations.

   Linux & macOS

   ```
   aws ssm describe-automation-executions \
       --filter Key=ExecutionId,Values=automation execution ID
   ```

   Windows

   ```
   aws ssm describe-automation-executions ^
       --filter Key=ExecutionId,Values=automation execution ID
   ```

   PowerShell

   ```
   Get-SSMAutomationExecutionList | `
       Where {$_.AutomationExecutionId -eq "automation execution ID"}
   ```

6. Pour afficher les détails de la progression de l'automatisation, exécutez la commande suivante. Remplacez chaque *automation execution ID* (ID d'exécution de l'automatisation) par vos propres informations.

   Linux & macOS

   ```
   aws ssm get-automation-execution \
       --automation-execution-id automation execution ID
   ```

Windows

```
aws ssm get-automation-execution ^
    --automation-execution-id automation execution ID
```

PowerShell

```
Get-SSMAutomationExecution `
    -AutomationExecutionId automation execution ID
```

Le système retourne des informations telles que les suivantes.

Linux & macOS

```
{
    "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 aussi surveiller le statut de l'automatisation dans la console. Dans la liste Automation executions (Exécutions Automation), sélectionnez l'exécution que vous venez juste d'exécuter, puis sélectionnez l'onglet Execution Steps (Étapes d'exécution). Cet onglet affiche le statut des actions de l'automatisation.

Mappage des cibles pour une automatisation

Utilisez le paramètre `Targets` afin de définir rapidement quelles ressources de votre flotte peuvent exécuter une automatisation. Par exemple, si vous souhaitez exécuter une automatisation qui redémarre vos instances gérées, au lieu de choisir manuellement des dizaines d'ID d'instance dans la console ou de les saisir dans une commande, vous pouvez cibler ces instances en spécifiant des balises Amazon Elastic Compute Cloud (Amazon EC2) avec le paramètre `Targets`.

Lorsque vous exécutez une automatisation qui utilise une cible, AWS Systems Manager crée une automatisation auxiliaire pour chaque cible. Par exemple, si vous ciblez les volumes Amazon Elastic Block Store (Amazon EBS) en spécifiant des balises, et que ces balises se résolvent en 100 volumes Amazon EBS, puis Systems Manager crée 100 automatisations enfants. L'automatisation parent est terminée lorsque toutes les automatisations enfants atteignent un état final.

> **Note**
>
> Toutes les valeurs `input parameters` que vous spécifiez au moment de l'exécution (dans la section Input Parameters (Paramètres d'entrée) de la console ou à l'aide de l'option `parameters` de l'interface de ligne de commande) sont automatiquement traitées par toutes les automatisations.

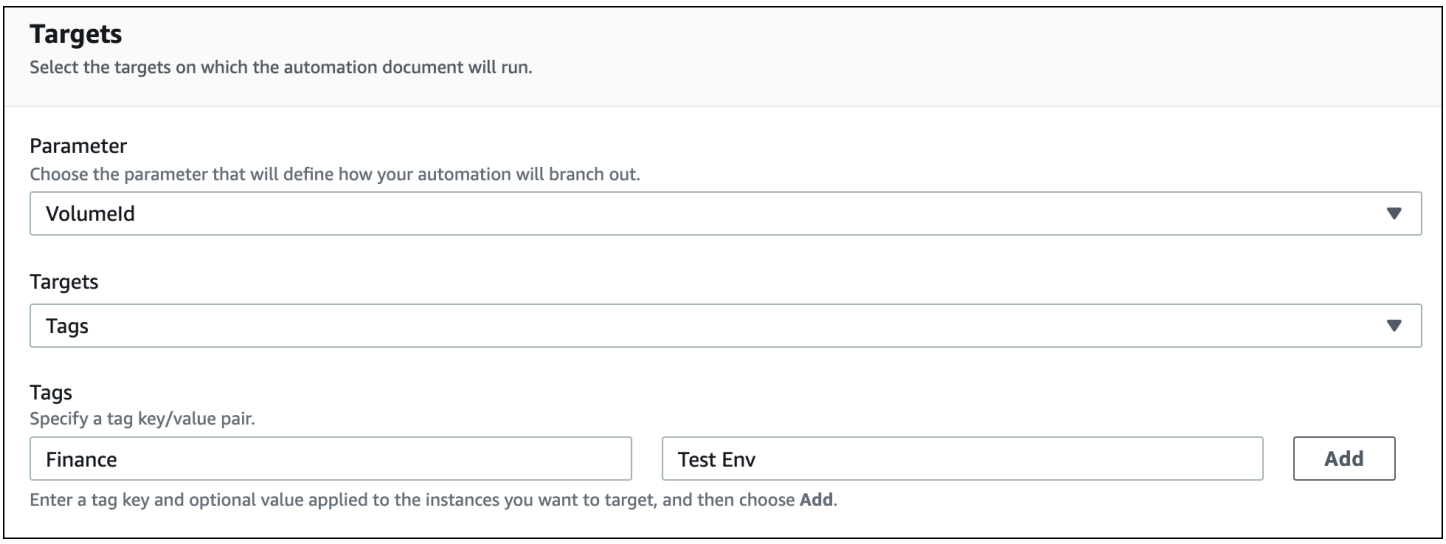
Pour cibler des ressources pour une automatisation, vous pouvez utiliser des balises, des groupes de ressources et des valeurs des paramètres. De plus, vous pouvez utiliser l'option `TargetMaps` pour cibler plusieurs valeurs de paramètre à l'aide de l'interface de ligne de commande ou d'un fichier. La section suivante décrit chacune de ces options de ciblage en détail.

Ciblage d'une balise

Vous pouvez spécifier une seule balise comme cible d'une automatisation. De nombreuses ressources AWS prennent en charge les balises, par exemple les instances Amazon Elastic Compute Cloud (Amazon EC2) et Amazon Relational Database Service (Amazon RDS), les volumes et instantanés Amazon Elastic Block Store (Amazon EBS), les Resource Groups et les compartiments Amazon Simple Storage Service (Amazon S3). Vous pouvez rapidement exécuter des automatisations sur vos ressources AWS en ciblant une balise. Une balise est une paire clé-valeur, telle que Operating_System:Linux (Système_Exploitation:Linux) ou Department:Finance (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 en tant que cible d'une automatisation, vous pouvez également spécifier un paramètre cible. Le paramètre cible utilise l'option `TargetParameterName`. En choisissant un paramètre cible, vous définissez le type de ressource sur lequel automatisation s'exécute. Le paramètre cible que vous spécifiez avec la balise doit être un paramètre valide défini dans le runbook. Par exemple, si vous souhaitez cibler des dizaines d'instances EC2 avec des balises, sélectionnez le paramètre cible `InstanceId`. En choisissant ce paramètre, vous définissez les instances comme type de ressource pour l'automatisation. Lors de la création d'un runbook personnalisé, vous devez définir le type de cible sur `/AWS::EC2::Instance`, de sorte que seules les instances soient utilisées. Dans le cas contraire, toutes les ressources portant la même balise seront ciblées. Lorsque vous ciblez des instances avec une balise, les instances résiliées peuvent être incluses.

La capture d'écran suivante utilise le runnbook `AWS-DetachEBSVolume`. Le paramètre cible logique est `VolumeId`.



Le runbook `AWS-DetachEBSVolume` inclut également une propriété spéciale appelée Target type (Type de cible), qui est définie sur `/AWS::EC2::Volume`. En d'autres termes, si la paire balise-clé `Finance:TestEnv` renvoie différents types de ressources (par exemple, des instances EC2, des volumes Amazon EBS, des instantanés Amazon EBS), seuls les volumes Amazon EBS sont utilisés.

Important

Les noms de paramètre cible sont sensibles à la casse. Si vous exécutez des automatisations en utilisant le AWS Command Line Interface (AWS CLI) ou AWS Tools for Windows PowerShell, vous devez saisir le nom du paramètre cible exactement tel qu'il est défini dans le runbook. Dans le cas contraire, le système renvoie une erreur `InvalidAutomationExecutionParametersException`. Vous pouvez utiliser l'opération DescribeDocumentAPI pour consulter des informations sur les paramètres cibles disponibles dans un runbook spécifique. Voici un exemple de AWS CLI commande qui fournit des informations sur le `AWS-DeleteSnapshot` document.

```
aws ssm describe-document \
    --name AWS-DeleteSnapshot
```

Voici quelques exemples de AWS CLI commandes qui ciblent les ressources à l'aide d'une balise.

Exemple 1 : ciblage d'une balise à l'aide d'une paire clé-valeur pour redémarrer les instances Amazon EC2

Cet exemple redémarre toutes les instances Amazon EC2 étiquetées avec la clé Department et la valeur de. HumanResources Le paramètre cible utilise le InstanceIdparamètre du runbook. 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 d'une balise à l'aide d'une paire clé-valeur pour supprimer les instantanés Amazon EBS

L'exemple suivant utilise le runbook `AWS-DeleteSnapshot` pour supprimer tous les instantanés avec la clé Name (Nom) et la valeur January2018Backups. Le paramètre cible utilise le VolumeIdparamètre.

```
aws ssm start-automation-execution \
    --document-name AWS-DeleteSnapshot \
    --targets Key=tag:Name,Values=January2018Backups \
    --target-parameter-name VolumeId
```

Ciblage AWS Resource Groups

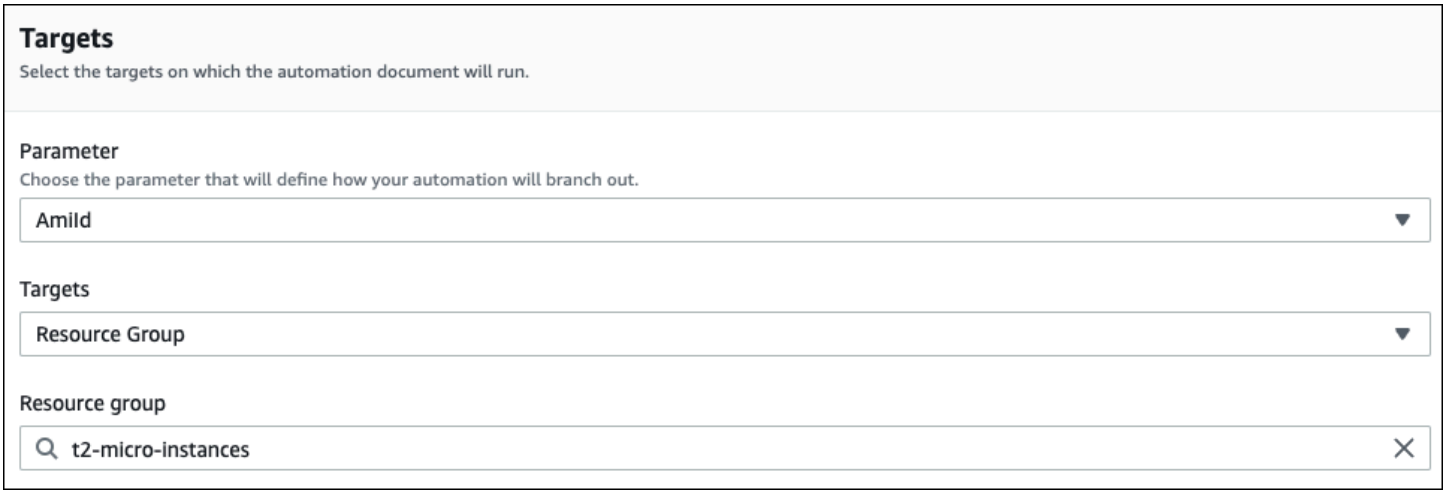
Vous pouvez spécifier un seul groupe de AWS ressources comme cible d'une automatisation. Systems Manager crée une automatisation enfant pour chaque objet du groupe de ressources cible.

Par exemple, supposons que l'un de vos groupes de ressources s'appelle PatchedAMIs. Ce groupe de ressources inclut une liste de 25 Amazon Machine Images (AMIs) Windows auxquelles des correctifs sont appliqués régulièrement. Si vous exécutez une automatisation qui utilise le runbook `AWS-CreateManagedWindowsInstance` et ciblez ce groupe de ressources, Systems Manager crée une automatisation enfant pour chacun des 25 AMIs. Autrement dit, en ciblant le groupe de ressources PatchedAMIs, l'automatisation crée 25 instances à partir d'une liste d' corrigées AMIs. L'automatisation parent se termine lorsque toutes les automatisations enfants arrivent au terme du traitement ou atteignent un état final.

La AWS CLI commande suivante s'applique à l'exemple de groupe de ressources Patchamis. La commande prend le AmiIdparamètre de l'`--target-parameter-name`option. Cette commande n'inclut pas d'autres paramètres pour définir le type d'instance à créer à partir de chaque AMI. Le runbook `AWS-CreateManagedWindowsInstance` renvoie par défaut au type d'instance t2.medium. Dans ce cas, la commande créerait donc 25 instances Amazon 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. Saisissez `ParameterValues` comme clé, puis la valeur de ressource spécifique où vous voulez que l'automatisation soit exécutée. Si vous spécifiez plusieurs valeurs, Systems Manager exécute une automatisation sur chaque valeur spécifiée.

Par exemple, supposons que le runbook comporte un paramètre InstanceID. Si vous ciblez les valeurs du paramètre InstanceID lorsque vous exécutez l'automatisation, Systems Manager exécute une automatisation enfant pour chaque valeur d'ID d'instance spécifiée. L'automatisation parent se termine lorsque l'automatisation a exécuté chaque instance spécifiée ou si l'exécution échoue. Vous pouvez cibler un maximum de 50 valeurs de paramètre.

L'exemple suivant utilise le runbook `AWS-CreateImage`. Le nom du paramètre cible spécifié est InstanceId. Les principales utilisations ParameterValues. Les valeurs sont deux ID d'instance Amazon EC2. 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**
>
> `AutomationAssumeRole` n'est pas un paramètre valide. Ne sélectionnez pas cet élément lors de l'exécution d'automatisation qui ciblent une valeur de paramètre.

Ciblage des mappages de valeurs des paramètres

L'option `TargetMaps` élargit votre capacité à cibler `ParameterValues`. Vous pouvez entrer une liste de valeurs de paramètres en utilisant `TargetMaps` au niveau de 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**
>
> L'option `TargetMaps` n'est pas prise en charge dans la console.

Utilisez le format suivant pour spécifier plusieurs valeurs de paramètre en utilisant l'option `TargetMaps` dans une commande. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations.

```
aws ssm start-automation-execution \
    --document-name runbook name \
    --target-maps “parameter=value, parameter 2=value, parameter 3=value”  “parameter
 4=value, parameter 5=value, parameter 6=value”
```

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.

```
[

    {“parameter”: "value", “parameter 2”: "value", “parameter 3”: "value"},

    {“parameter 4”: "value", “parameter 5”: "value", "parameter 6": "value"}

]
```

Enregistrez le fichier avec l'extension de fichier .json. Vous pouvez appeler le fichier en utilisant la commande suivante. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

```
aws ssm start-automation-execution \
    --document-name runbook name \
    --parameters input parameters \
    --target-maps path to file/file name.json
```

Vous pouvez également télécharger le fichier à partir d'un compartiment Amazon Simple Storage Service (Amazon S3), dans la mesure où vous avez l'autorisation de lire les données de ce compartiment. Utilisez le format de commande suivant. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

```
aws ssm start-automation-execution \
    --document-name runbook name \
    --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 Amazon EC2 de différents types à partir d'AMIs distinctes. Pour exécuter cette tâche, l'utilisateur crée un runbook nommé AMI_Testing. Ce runbook définit deux paramètres d'entrée : `instanceType` et `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"],
```

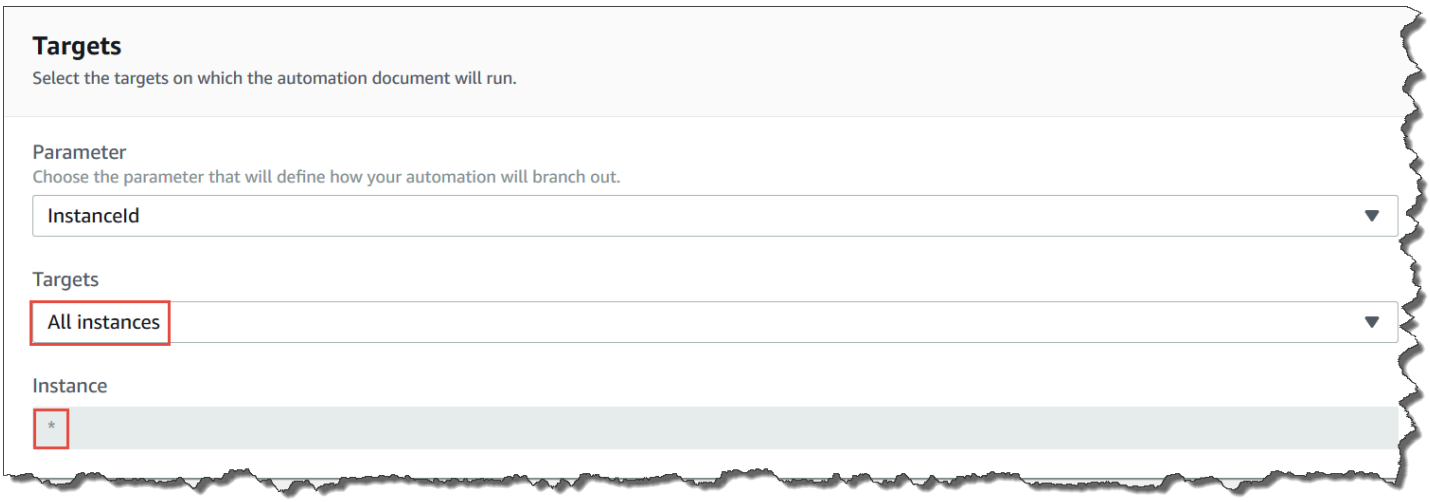
```
    "imageId" : ["ami-b70554c8"]
  },
  {
    "instanceType" : ["t2.medium"],
    "imageId" : ["ami-cfe4b2b0"]
  },
  {
    "instanceType" : ["t2.medium"],
    "imageId" : ["ami-cfe4b2b0"]
  },
  {
    "instanceType" : ["t2.medium"],
    "imageId" : ["ami-cfe4b2b0"]
  }
]
```

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 Amazon EC2

Vous pouvez exécuter une automatisation sur toutes les instances Amazon EC2 actuelles Compte AWS en Région AWS choisissant Toutes les instances dans la liste des cibles. Par exemple, si vous souhaitez redémarrer toutes les instances Amazon EC2, les vôtres Compte AWS et les instances actuelles Région AWS, vous pouvez choisir le **`AWS-RestartEC2Instance`** runbook, puis toutes les instances dans la liste des cibles.



Après que vous avez choisi All instances (Toutes les instances), Systems Manager remplit le champ Instance avec un astérisque (*) et empêche la modification de ce champ (qui est alors grisé). Systems Manager rend également le InstanceIdchamp du champ Paramètres d'entrée indisponible pour les modifications. Rendre ces champs indisponibles pour empêcher leur modification est le comportement attendu si vous choisissez de cibler toutes les instances.

Automatisations de contrôle à grande échelle

Vous pouvez contrôler le déploiement d'une automatisation dans l'ensemble d'une flotte de ressources AWS en spécifiant une valeur de simultanéité et un seuil d'erreurs. La simultanéité et le seuil d'erreurs sont appelés collectivement contrôles du débit.

Simultanéité

Utilisez Concurrency (Simultanéité) pour spécifier combien de ressources sont autorisées à exécuter une automatisation simultanément. Elle permet de limiter l'impact sur vos ressources ou les temps d'arrêt 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 chaque fois plus de ressources jusqu'à ce que la valeur de simultanéité soit atteinte.

Seuils d'erreurs

Utilisez un seuil d'erreurs pour spécifier le nombre d'automatisations qui peuvent échouer jusqu'à ce que AWS 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 trois erreurs, le système cesse d'envoyer l'exécution de l'automatisation à la réception de la quatrième erreur. Si vous spécifiez 0, le système cesse d'exécuter l'automatisation à des cibles supplémentaires une fois que le premier résultat d'erreur est renvoyé.

De même, si vous envoyez une automatisation à 50 instances et que vous définissez le seuil d'erreurs sur la valeur 10 %, le système arrête d'envoyer la commande aux instances supplémentaires à la réception de la cinquième erreur. Les appels qui exécutent déjà une automatisation quand un seuil d'erreurs est atteint sont autorisés à se terminer, mais certaines de ces automatisations peuvent également échouer. Pour vous assurer qu'il n'y aura pas plus d'erreurs que le nombre spécifié pour le seuil d'erreurs, définissez la valeur Concurrency sur 1 afin que les automatisations s'exécutent une à une.

Exécution d'automatisations dans plusieurs régions et comptes Régions AWS

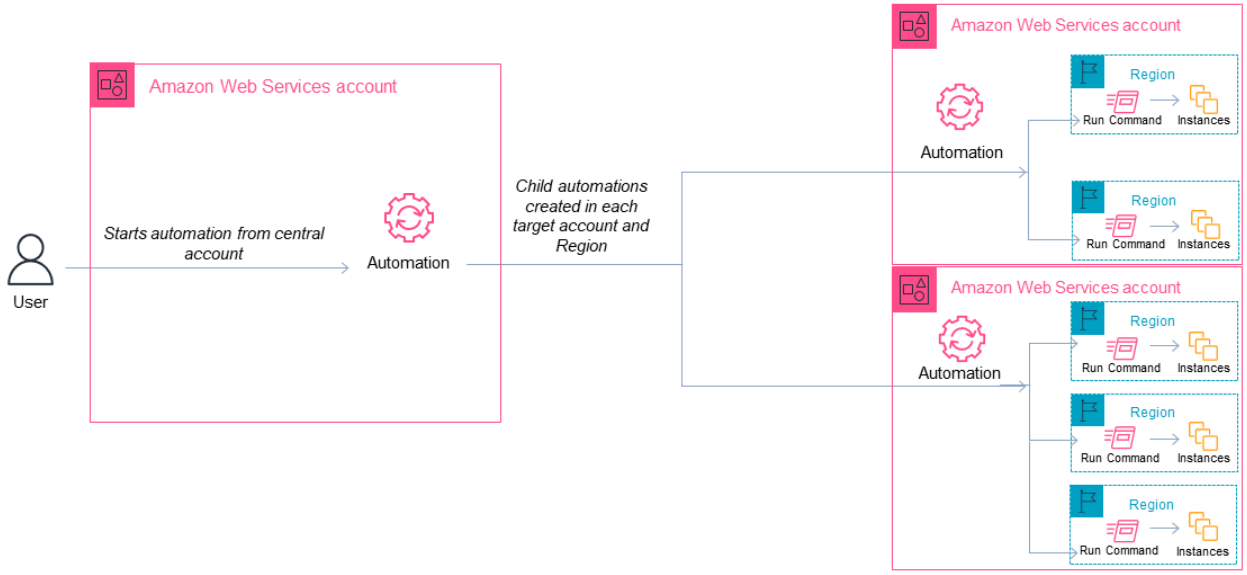
L'exécution des automatisations AWS Systems Manager sur plusieurs Régions AWS et Comptes AWS ou AWS Organizations unités organisationnelles à partir d'un compte central. Automation est une fonctionnalité de AWS Systems Manager. Exécuter des automatisations dans plusieurs régions et comptes ou unités organisationnelles permet de réduire le temps nécessaire pour administrer vos ressources AWS, tout en renforçant la sécurité de votre environnement informatique.

Par exemple, vous pouvez effectuer les opérations suivantes à l'aide de runbooks Automation :

- Implémentation des correctifs ainsi que des mises à jour de sécurité de manière centralisée.
- Correction des écarts de conformité des configurations de VPC ou des politiques de compartiment Amazon S3.
- Gestion des ressources, telles que des instances Amazon Elastic Compute Cloud (Amazon EC2) EC2 à grande échelle.

Le graphique suivant illustre un exemple d'utilisateur exécutant le runbook `AWS-RestartEC2Instances` dans plusieurs régions et comptes à partir d'un compte central. L'automatisation localise les instances à l'aide des balises définies dans les régions et les comptes ciblés.

> **Note**
>
> Lorsque vous exécutez l'automatisation dans plusieurs régions et comptes, vous ciblez les ressources à l'aide de balises ou du nom d'un groupe de ressources AWS. Le groupe de ressources doit exister dans chaque compte et chaque région cibles. Le nom du groupe de ressources doit être le même dans chaque compte et dans chaque région ciblés. L'automatisation ne peut pas s'exécuter sur les ressources pour lesquelles aucune balise n'a été spécifiée ou qui ne sont pas incluses dans le groupe de ressources défini.



Choisissez un compte central pour Automation

Si vous souhaitez exécuter des automatisations sur plusieurs unités d'organisation, le compte central doit disposer des autorisations nécessaires afin de répertorier tous les comptes dans les unités d'organisation. Cette exécution n'est possible qu'à partir d'un compte administrateur délégué ou du compte de gestion de l'organisation. Nous vous recommandons de suivre les bonnes pratiques AWS Organizations et d'utiliser un compte administrateur délégué. Pour de plus amples informations sur les bonnes pratiques AWS Organizations, consultez Bonnes pratiques relatives au compte de gestion dans le Guide de l'utilisateur AWS Organizations. Pour créer un compte d'administrateur délégué

pour Systems Manager, utilisez la commande `register-delegated-administrator` avec la AWS CLI comme indiqué dans l'exemple suivant.

```
aws organizations register-delegated-administrator \
    --account-id delegated admin account ID \
    --service-principal ssm.amazonaws.com
```

Si vous souhaitez exécuter des automatisations sur plusieurs comptes non gérés par AWS Organizations, nous vous recommandons de créer un compte dédié à la gestion de l'automatisation. L'exécution de toutes les automatisations inter-comptes à partir d'un compte dédié simplifie la gestion des autorisations IAM, les efforts de dépannage et crée une couche de séparation entre les opérations et l'administration. Cette approche est également recommandée si vous utilisez AWS Organizations, mais souhaite uniquement cibler des comptes individuels et non des unités d'organisation.

Le fonctionnement de l'exécution des automatisations

L'exécution d'automatisations dans plusieurs régions et comptes ou unités organisationnelles fonctionne comme suit :

1. Vérifiez que toutes les ressources sur lesquelles vous souhaitez exécuter l'automatisation, dans l'ensemble des régions et comptes ou unités organisationnelles, 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 plus d'informations, consultez Que sont les groupes de ressources? dans le AWS Resource Groups le guide de l'utilisateur et des balises.
2. Connectez-vous au compte que vous souhaitez configurer en tant que compte central de l'automatisation.
3. Utilisez le Configuration des autorisations du compte de gestion pour l'exécution d'automatisations entre plusieurs régions et plusieurs comptes procédure décrite dans cette rubrique afin de créer le second role IAM:
   - **`AWS-SystemsManager-AutomationAdministrationRole`** - Ce rôle donne à l'utilisateur l'autorisation d'exécuter des automatisations dans plusieurs comptes et unités organisationnelles.
   - **`AWS-SystemsManager-AutomationExecutionRole`** - Ce rôle donne à l'utilisateur l'autorisation d'exécuter des automatisations dans plusieurs comptes et unités organisationnelles.

4. Sélectionnez le runbook, les régions et les comptes ou unités organisationnelles dans lesquels vous souhaitez exécuter l'automatisation.

> **Note**
>
> Les automatisations ne s'exécutent pas de manière récursive via les unités d'organisation. Assurez-vous que l'unité d'organisation cible contient les comptes souhaités. Si vous sélectionnez un runbook personnalisé, il doit être partagé avec tous les comptes cibles. Pour obtenir des informations sur le partage des runbooks, consultez Partage de documents SSM. Pour obtenir des informations sur l'utilisation de runbooks partagés, consultez Utilisation de documents SSM partagés.

5. Exécutez l'automatisation.

> **Note**
>
> Lorsque vous exécutez des automatisations sur plusieurs régions, comptes ou unités organisationnelles, l'automatisation que vous exécutez à partir du compte principal démarre des automatisations enfants dans chacun des comptes cibles. L'automatisation dans le compte principal comprendra des étapes `aws:executeAutomation` pour chacun des comptes cibles. Si vous démarrez une automatisation à partir de nouvelles régions lancées après le 20 mars 2019 et que vous ciblez une région activée par défaut, l'automatisation échoue. Si vous démarrez une automatisation à partir d'une région activée par défaut et que vous ciblez une région que vous avez activée, l'automatisation s'exécute correctement.

6. Utilisez les opérations d'API GetAutomationExecution, DescribeAutomationStepExecutions, et DescribeAutomationExecutions à partir de la console AWS Systems Manager ou de la AWS CLI afin de contrôler la progression de l'automatisation. La sortie des étapes de l'automatisation dans votre compte principal sera l'`AutomationExecutionId` des automations enfants. Pour afficher la sortie des automatisations enfants créées dans vos comptes cibles, spécifiez bien le compte, la région et `AutomationExecutionId` correspondants, dans votre demande.

Configuration des autorisations du compte de gestion pour l'exécution d'automatisations entre plusieurs régions et plusieurs comptes

Utilisez la procédure suivante pour créer les rôles IAM requis pour l'exécution de Systems Manager Automation dans plusieurs régions et plusieurs comptes avec AWS CloudFormation. Cette procédure

décrit la création du rôle **AWS-SystemsManager-AutomationAdministrationRole**. Créez uniquement ce rôle dans le compte central d'automatisation. Cette procédure décrit également la création du rôle **AWS-SystemsManager-AutomationExecutionRole**. Vous devez créer ce rôle dans chaque compte que vous souhaitez cibler pour exécuter les automatisations dans plusieurs régions et plusieurs comptes. Nous vous recommandons d'utiliser AWS CloudFormation StackSets pour créer le rôle **AWS-SystemsManager-AutomationExecutionRole** dans les comptes que vous voulez cibler pour exécuter des automatisations dans plusieurs régions et plusieurs comptes.

Pour créer les rôles IAM requis pour les automatisations multi-régions et multi-comptes à l'aide de AWS CloudFormation

1. Téléchargez et compressez le AWS-SystemsManager-AutomationAdministrationRole.zip. Ou, si vos comptes sont gérés par AWS Organizations AWS-SystemsManager-AutomationAdministrationRole (org).zip. Ce fichier contient le fichier de modèle AWS CloudFormation `AWS-SystemsManager-AutomationAdministrationRole.yaml`.
2. Ouvrez la console AWS CloudFormation à l'adresse https://console.aws.amazon.com/cloudformation.
3. Sélectionnez Créer une pile.
4. Dans la section Specify template (Spécifier un modèle), sélectionnez Upload a template (Charger un modèle).
5. Sélectionnez Choose file (Choisir un fichier) et sélectionnez le fichier modèle AWS CloudFormation `AWS-SystemsManager-AutomationAdministrationRole.yaml`.
6. Sélectionnez Suivant.
7. Dans la section Spécifier les détails, entrez un nom dans le champ Nom de la pile.
8. Choisissez Next (Suivant).
9. Sur la page Configure stack options (Configurer les options de pile), saisissez des valeurs pour les options que vous souhaitez utiliser. Sélectionnez Suivant.
10. Dans la page Review (Vérification), défilez vers le bas et choisissez l'option I acknowledge that AWS CloudFormation might create IAM resources with custom names (Je reconnais que CFN peut créer des ressources IAM avec des noms personnalisés).
11. Sélectionnez Créer une pile.

AWS CloudFormation affiche le statut CREATE_IN_PROGRESS pendant environ trois minutes. Le statut passe à CREATE_COMPLETE.

Répétez la procédure suivante dans chaque compte que vous souhaitez cibler pour exécuter les automatisations multi-régions et multi-comptes.

Pour créer les rôles IAM requis pour les automatisations multi-régions et multi-comptes à l'aide de AWS CloudFormation

1. Téléchargez AWS-SystemsManager-AutomationExecutionRole.zip. Ou, si vos comptes sont gérés par AWS Organizations AWS-SystemsManager-AutomationExecutionRole(org).zip. Ce fichier contient le fichier de modèle AWS CloudFormation AWS-SystemsManager-AutomationExecutionRole.yaml.
2. Ouvrez la console AWS CloudFormation à l'adresse https://console.aws.amazon.com/cloudformation.
3. Sélectionnez Créer une pile.
4. Dans la section Specify template (Spécifier un modèle), sélectionnez Upload a template (Charger un modèle).
5. Sélectionnez Choose file (Choisir un fichier) et sélectionnez le fichier modèle AWS CloudFormation AWS-SystemsManager-AutomationExecutionRole.yaml.
6. Sélectionnez Suivant.
7. Dans la section Spécifier les détails, entrez un nom dans le champ Nom de la pile.
8. Dans la section Parameters (Paramètres), dans le champ AdminAccountId, saisissez l'identifiant du compte central Automation.
9. Si vous configurez ce rôle pour un environnement AWS Organizations, il existe un autre champ dans la section appelée OrganizationID. Entrez l'identifiant de votre organisation AWS.
10. Choisissez Next (Suivant).
11. Sur la page Configure stack options (Configurer les options de pile), saisissez des valeurs pour les options que vous souhaitez utiliser. Sélectionnez Suivant.
12. Dans la page Review (Vérification), défilez vers le bas et choisissez l'option I acknowledge that AWS CloudFormation might create IAM resources with custom names (Je reconnais que CFN peut créer des ressources IAM avec des noms personnalisés).
13. Sélectionnez Créer une pile.

AWS CloudFormation affiche le statut CREATE_IN_PROGRESS pendant environ trois minutes. Le statut passe à CREATE_COMPLETE.

Exécuter une automatisation dans plusieurs comptes et régions (console)

La procédure suivante décrit comment utiliser la console Systems Manager 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 :

- L'utilisateur ou le rôle que vous utilisez pour exécuter Automation dans plusieurs régions ou plusieurs comptes doit disposer de l'autorisation `iam:PassRole` pour le rôle `AWS-SystemsManager-AutomationAdministrationRole`.
- ID de Compte AWS ou unités organisationnelles dans lesquels vous souhaitez exécuter l'automatisation.
- 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, sélectionnez Automation (Automatisation), puis Execute automation (Exécuter l'automatisation).
3. Dans la liste Automation document (Document Automation), sélectionnez un runbook. Sélectionnez une ou plusieurs options dans le panneau Document categories (Catégories de documents) pour filtrer les documents SSM en fonction de leur but. Pour afficher un runbook vous appartenant, sélectionnez l'onglet Owned by me (M'appartenant). Pour afficher un runbook partagé avec votre compte, sélectionnez l'onglet Shared with me (Partagé avec moi). Pour afficher tous les runbooks, sélectionnez l'onglet All documents (Tous les documents).

   > **Note**
   >
   > Vous pouvez consulter les informations sur un runbook en sélectionnant 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 :

- Version par défaut lors de l’exécution : sélectionnez cette option si le runbook d’automatisation est mis à jour régulièrement et qu’une nouvelle version par défaut est attribuée.
- Dernière version lors de l’exécution : sélectionnez cette option si le runbook d’automatisation est mis à jour régulièrement et que vous souhaitez exécuter la dernière version mise à jour.
- 1 (Par défaut) : sélectionnez cette option pour exécuter la première version du document, qui est la version par défaut.

5. Choisissez Next (Suivant).
6. Sur la page Execute automation document (Exécuter le document d'Automation) , sélectionnez Multi-account and Region (Compte et région multiples).
7. Dans la section Target accounts and Regions (Comptes de destination et régions), utilisez le champ Accounts and organizational (Comptes et UO) pour spécifier les différents comptes Comptes AWS ou unités organisationnelles AWS dans lesquels vous souhaitez exécuter l'automatisation. Séparez les différents comptes ou unités organisationnelles par une virgule.
8. Utilisez la liste Régions AWS pour choisir une ou plusieurs régions dans lesquelles vous souhaitez exécuter l'automatisation.
9. Utilisez les options Multi-Region and account rate control (Plusieurs régions et contrôle du débit du compte) pour restreindre l'exécution d'automatisations à un nombre limité de comptes qui s'exécutent dans un nombre limité de régions. Ces options ne limitent pas le nombre de ressources AWS qui peuvent exécuter les automatisations.

    a. Dans la section Location (account-Region pair) concurrency (Simultanéité de l'emplacement [paire compte/région]), sélectionnez une option pour restreindre 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) Comptes AWS qui sont situés dans quatre (4) Régions AWS, puis que Systems Manager exécute des automatisations dans un total de 20 paires de comptes-régions. Vous pouvez utiliser cette option pour spécifier un nombre absolu, tel que **2**, de manière à ce que l'automatisation s'exécute uniquement dans 2 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 5 paires compte-région.

        - Sélectionnez targets (cibles) pour saisir un nombre absolu de paires compte-région pouvant exécuter l'automatisation simultanément.

- Sélectionnez percentage (pourcentage) pour saisir un pourcentage du nombre total de paires compte-région pouvant exécuter l'automatisation simultanément.

b. Dans la section Error threshold (Seuil d'erreurs), sélectionnez une option :

- Sélectionnez errors (erreurs) pour indiquer un nombre absolu d'erreurs autorisées avant qu'Automation ne cesse d'envoyer l'automatisation à d'autres ressources.
- Sélectionnez percentage (pourcentage) pour indiquer un pourcentage d'erreurs autorisées avant qu'Automation ne cesse d'envoyer l'automatisation à d'autres ressources.

10. Dans la section Targets (Cibles), sélectionnez 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 runbook Automation 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 choisissez de cibler des ressources à l'aide de valeurs de paramètre, saisissez la valeur de paramètre du paramètre que vous avez choisi, dans la section Input parameters (Paramètres d'entrée).

ii. Si vous choisissez de cibler les ressources avec AWS Resource Groups, sélectionnez 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 runbook Automation sur toutes les instances du Compte AWS et de la Région AWS actuels, sélectionnez All instances (Toutes les instances).

11. Dans la section Input parameters (Paramètres d'entrée), spécifiez les entrées obligatoires. Choisissez le rôle de service IAM `AWS-SystemsManager-AutomationAdministrationRole` dans la liste AutomationAssumeRole.

> **Note**
>
> Vous pourriez ne pas avoir 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 runbook `AWS-RestartEC2Instance`, vous n'avez pas à spécifier ou à choisir d'ID d'instance dans la section Inpunt parameters (Paramètres d'entrée). L'automatisation localise les instances à redémarrer en utilisant les balises que vous avez spécifiées.

12. (Facultatif) Choisissez une alarme CloudWatch à appliquer à votre automatisation à des fins de surveillance. Afin d'associer une alarme CloudWatch à votre automatisation, le principal IAM démarrant l'automatisation doit disposer de l'autorisation pour l'action `iam:createServiceLinkedRole`. Pour de plus amples informations relatives à la configuration des alarmes CloudWatch, consultez Utilisation des alarmes Amazon CloudWatch. Veuillez noter que l'activation de votre alarme entraîne l'annulation de l'automatisation et l'exécution des étapes `OnCancel` définies. Si vous utilisez AWS CloudTrail, vous verrez l'appel d'API dans votre journal.
13. 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 Simultanéité, sélectionnez une option :

    - Sélectionnez targets (cibles) pour entrer un nombre absolu de cibles pouvant exécuter le flux de travail Automation simultanément.
    - Sélectionnez percentage (pourcentage) pour indiquer un pourcentage de l'ensemble de cibles pouvant exécuter le flux de travail Automation simultanément.
14. Dans la section Error threshold (Seuil d'erreurs), sélectionnez une option :

    - Sélectionnez erreurs pour indiquer un nombre absolu d'erreurs autorisées avant qu'Automation ne cesse d'envoyer le flux de travail à d'autres ressources.
    - Sélectionnez pourcentage pour indiquer un pourcentage d'erreurs autorisées avant qu'Automation ne cesse d'envoyer le flux de travail à d'autres ressources.
15. Sélectionnez Execute (Exécuter).

Exécuter Automation dans plusieurs comptes et régions (ligne de commande)

La procédure suivante décrit comment utiliser l'AWS CLI (sous Linux ou Windows) ou les AWS Tools for 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 :

- ID de Compte AWS ou unités organisationnelles dans lesquels vous souhaitez exécuter l'automatisation.
- 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 AWS Tools for PowerShell si vous ne l'avez pas déjà fait.

   Pour plus d'informations, consultez la section Installation ou mise à jour de la version la plus récente de l'AWS CLI et Installation d'AWS Tools for PowerShell.
2. Utilisez le format suivant pour créer une commande permettant d'exécuter une automatisation dans plusieurs régions et comptes. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   Linux & macOS

   ```
   aws ssm start-automation-execution \
           --document-name runbook name \
           --parameters AutomationAssumeRole=arn:aws:iam::management account
    ID:role/AWS-SystemsManager-AutomationAdministrationRole \
           --target-parameter-name parameter name \
           --targets Key=tag key,Values=value \
           --target-locations Accounts=account ID,account ID
    2,Regions=Region,Region 2,ExecutionRoleName=AWS-SystemsManager-
   AutomationExecutionRole
   ```

   Windows

   ```
   aws ssm start-automation-execution ^
           --document-name runbook name ^
           --parameters AutomationAssumeRole=arn:aws:iam::management account
    ID:role/AWS-SystemsManager-AutomationAdministrationRole ^
           --target-parameter-name parameter name ^
           --targets Key=tag key,Values=value ^
   ```

```
        --target-locations Accounts=account ID,account ID
 2,Regions=Region,Region 2,ExecutionRoleName=AWS-SystemsManager-
AutomationExecutionRole
```

PowerShell

```
$Targets = New-Object Amazon.SimpleSystemsManagement.Model.Target
    $Targets.Key = "tag key"
    $Targets.Values = "value"

    Start-SSMAutomationExecution `
        -DocumentName "runbook name" `
        -Parameter @{
        "AutomationAssumeRole"="arn:aws:iam::management account ID:role/AWS-
SystemsManager-AutomationAdministrationRole" } `
        -TargetParameterName "parameter name" `
        -Target $Targets `
        -TargetLocation @{
        "Accounts"="account ID","account ID 2";
        "Regions"="Region","Region 2";
        "ExecutionRoleName"="AWS-SystemsManager-AutomationExecutionRole" }
```

Voici quelques exemples.

Exemple 1 : cet exemple redémarre les instances EC2 dans les comptes `123456789012` et `987654321098`, qui sont situés dans les régions `us-east-2` et `us-west-1`. Les instances doivent être balisées avec la valeur de paire de clés de balise `Env-PROD`.

Linux & macOS

```
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 dans les comptes `123456789012` et `987654321098`, qui sont situés dans la région `eu-central-1`. Les instances doivent être membres du groupe de ressources AWS `prod-instances`.

Linux & macOS

```
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 organisation AWS `ou-1a2b3c-4d5e6c`. 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 & macOS

```
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 & macOS

```
{
```

```
        "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 des informations détaillées relatives à l'automatisation. Remplacez *automation execution ID* (ID d'exécution de l'automatisation) par vos propres informations.

   Linux & macOS

   ```
   aws ssm describe-automation-executions \
           --filters Key=ExecutionId,Values=automation execution ID
   ```

   Windows

   ```
   aws ssm describe-automation-executions ^
           --filters Key=ExecutionId,Values=automation execution ID
   ```

   PowerShell

   ```
   Get-SSMAutomationExecutionList | `
           Where {$_.AutomationExecutionId -eq "automation execution ID"}
   ```

4. Exécutez la commande suivante pour afficher des informations détaillées relatives à l'automatisation.

   Linux & macOS

   ```
   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 aussi surveiller le statut de l'automatisation dans la console. Dans la liste Automation executions (Exécutions Automation), sélectionnez l'exécution que vous venez juste d'exécuter, puis sélectionnez l'onglet Execution Steps (Étapes d'exécution). Cet onglet affiche le statut des actions de l'automatisation.

Plus d'informations

[Correctif centralisé à plusieurs comptes et plusieurs régions avec AWS Systems Manager Automation]

Exécution d'automatisations basées sur les événements

Vous pouvez démarrer une automatisation en spécifiant un runbook comme cible d'un EventBridge événement Amazon. Vous pouvez démarrer des automatisations selon un programme ou lorsqu'un événement de système AWS spécifique se produit. Supposons, par exemple, que vous créiez un runbook nommé BootStrapInstancesqui installe le logiciel sur une instance au démarrage de celle-ci. Pour spécifier le BootStrapInstancesrunbook (et l'automatisation correspondante) comme 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 modification d'état d'instance EC2, État(s) spécifique(s): en cours exécution, Toute instance.) Vous devez ensuite utiliser les procédures suivantes pour spécifier le BootStrapInstancesrunbook comme cible de l'événement à l'aide de la EventBridge console et AWS Command Line Interface (AWS CLI). Lors du démarrage d'une nouvelle instance, le système exécute l'automatisation et installe le logiciel.

Pour plus d'informations sur la création de runbooks, consultez Créer vos propres runbooks.

Création d'un EventBridge événement utilisant un runbook (console)

Utilisez la procédure suivante pour configurer un runbook comme cible d'un EventBridge événement.

Pour configurer un runbook en tant que cible d'une règle d' EventBridge événement

1. Ouvrez la EventBridge console Amazon à l'adresse https://console.aws.amazon.com/events/.
2. Dans le volet de navigation, choisissez Règles.
3. Choisissez Créer une règle.
4. 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.
5. Pour Event bus (Bus d'événement), sélectionnez le bus d'événement que vous souhaitez associer à cette règle. Si vous souhaitez que cette règle réponde aux événements correspondants qui proviennent des vôtres Compte AWS, sélectionnez par défaut. Lorsqu'un événement Service AWS de votre compte est émis, il est toujours redirigé vers le bus d'événements par défaut de votre compte.
6. Choisissez comment la règle est déclenchée.

Pour créer une règle basée sur…	Faites ceci...	
Événement	a. Pour Rule type (Type de règle), choisissez Rule with an event pattern (Règle avec un modèle d'événement). b. Choisissez Suivant. c. Dans Source de l'événement, sélectionnez AWS événements ou événements EventBridge partenaires.	

Pour créer une règle basée sur…	Faites ceci...	
	d. Pour la section Event pattern (Modèle d'événement), effectuez l'une des opérations suivantes : • Pour utiliser un modèle afin de créer votre modèle d'événement, sélectionnez Formulaire de modèle d'événement et choisissez Source de l'événement, AWS Service, et Type d'évènement. Si vous choisissez Tous les événements comme type d'événement, tous les événements émis par le Service AWS seront conformes à la règle. Pour personnaliser le modèle, choisissez Custom pattern (JSON editor) (Modèle personnalisé [éditeur JSON]) et effectuez vos modifications. • Pour utiliser un modèle d'événement personnalisé, choisissez Custom pattern (JSON editor) (Modèle personnal	

Pour créer une règle basée sur…	Faites ceci...	
	isé [éditeur JSON]) et créez votre modèle d'événement.	

Pour créer une règle basée sur…	Faites ceci...	
Planificateur	a. Pour Rule type (Type de règle), choisissez Schedule (Planifier). b. Choisissez Suivant. c. Pour Schedule pattern (Planifier le modèle), effectuez l'une des étapes suivantes : • Pour utiliser une expression cron pour définir la planification, choisissez A fine-grai ned schedule that runs at a specific time, such as 8:00 a.m. (Un programme détaillé qui s'exécute à une heure précise, par exemple 8 h). PST on the first Monday of every month (PST le premier du lundi de chaque mois) et saisissez l'expression cron. • Pour utiliser une expression de rythme pour définir la planifica tion, choisissez un programme qui fonctionne à un rythme régulier, par exemple toutes les 10 minutes et	

Pour créer une règle basée sur…	Faites ceci...	
	saisissez l'expression de rythme.	

7. Choisissez Suivant.
8. Pour Types de cibles, choisissez service AWS .
9. Pour Select a target (Sélectionner une cible), choisissez Systems Manager Automation.
10. Pour Document, sélectionnez un runbook à utiliser lorsque votre cible est appelée.
11. Dans la section Configure automation parameter(s) (Configurer le(s) paramètre(s) d'automatisation), conservez les valeurs de paramètre par défaut (si disponibles) ou saisissez vos propres valeurs.

> **Note**
>
> Pour créer une cible, vous devez spécifier une valeur pour chaque paramètre requis. Si vous ne le faites pas, le système crée la règle mais elle ne s'exécute pas.

12. Pour de nombreux types de cibles, EventBridge nécessite des autorisations pour envoyer des événements à la cible. Dans ces cas, EventBridge vous pouvez créer le rôle IAM nécessaire à l'exécution de votre règle. Effectuez l'une des actions suivantes :

    - Pour créer un rôle IAM automatiquement, sélectionnez Créer un rôle pour cette ressource spécifique.
    - Pour utiliser un rôle IAM que vous avez créé précédemment, choisissez Use existing role (Utiliser un rôle existant) et sélectionnez le rôle existant dans la liste déroulante. Notez que vous devrez peut-être mettre à jour la politique de confiance pour inclure EventBridge votre rôle IAM. Voici un exemple :

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

```
                    "events.amazonaws.com",
                    "ssm.amazonaws.com"
                ]
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

13. Choisissez Suivant.
14. (Facultatif) Saisissez une ou plusieurs balises pour la règle. Pour plus d'informations, consultez la section Marquage de vos EventBridge ressources Amazon dans le guide de l' EventBridge utilisateur Amazon.
15. Choisissez Suivant.
16. Consultez les détails de la règle et choisissez Create rule (Créer une règle).

Création d'un EventBridge événement utilisant un runbook (ligne de commande)

La procédure suivante décrit comment utiliser AWS CLI (sous Linux ou Windows) ou comment AWS Tools for PowerShell créer une règle d' EventBridge événement et configurer un runbook comme cible.

Pour configurer un runbook en tant que cible d'une règle d' EventBridge événement

1. Installez et configurez le AWS CLI ou le AWS Tools for PowerShell, si ce n'est pas déjà fait.

   Pour plus d'informations, consultez la section Installation ou mise à jour de la version la plus récente de l' AWS CLI et Installation d' AWS Tools for PowerShell.
2. Créez une commande pour spécifier une nouvelle règle EventBridge d'événement. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   Déclencheurs basés sur un calendrier

   Linux & macOS

   ```
   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 d' EventBridge événement qui commence tous les jours à 9 h 00 (UTC).

Linux & macOS

```
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 & macOS

```
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 d' EventBridge événement qui démarre lorsqu'une instance EC2 de la région change d'état.

Linux & macOS

```
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 les détails de la nouvelle EventBridge règle de la manière suivante.

Linux & macOS

```
{
"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 runbook comme cible de la règle d' EventBridge événement que vous avez créée à l'étape 2. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

Linux & macOS

```
aws events put-targets \
--rule rule name \
--targets '{"Arn": "arn:aws:ssm:region:account ID:automation-definition/runbook
 name","Input":"{\"input parameter\":[\"value\"],\"AutomationAssumeRole\":
[\"arn:aws:iam::123456789012:role/AutomationServiceRole\"]}","Id": "target
 ID","RoleArn": "arn:aws:iam::123456789012:role/service-role/EventBridge service
 role"}'
```

Windows

```
aws events put-targets ^
--rule rule name ^
```

```
--targets '{"Arn": "arn:aws:ssm:region:account ID:automation-definition/runbook
 name","Input":"{\"input parameter\":[\"value\"],\"AutomationAssumeRole\":
[\"arn:aws:iam::123456789012:role/AutomationServiceRole\"]}","Id": "target
 ID","RoleArn": "arn:aws:iam::123456789012:role/service-role/EventBridge service
 role"}'
```

PowerShell

```
$Target = New-Object Amazon.CloudWatchEvents.Model.Target
$Target.Id = "target ID"
$Target.Arn = "arn:aws:ssm:region:account ID:automation-definition/runbook name"
$Target.RoleArn = "arn:aws:iam::123456789012:role/service-role/EventBridge
 service role"
$Target.Input = '{"input parameter":["value"],"AutomationAssumeRole":
["arn:aws:iam::123456789012:role/AutomationServiceRole"]}'

Write-CWETarget `
-Rule "rule name" `
-Target $Target
```

L'exemple suivant crée une cible d' EventBridge événement qui démarre l'ID d'instance spécifié à l'aide du runbookAWS-StartEC2Instance.

Linux & macOS

```
aws events put-targets \
--rule DailyAutomationRule \
--targets '{"Arn": "arn:aws:ssm:region:*: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:region:*: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:region:*: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 retourne des informations telles que les suivantes.

Linux & macOS

```
{
"FailedEntries": [],
"FailedEntryCount": 0
}
```

Windows

```
{
"FailedEntries": [],
"FailedEntryCount": 0
}
```

PowerShell

Il n'y a aucune sortie si la commande aboutit pour PowerShell.

Exécution manuelle d'une automatisation

Les procédures suivantes expliquent comment utiliser la console AWS Systems Manager et l'AWS Command Line Interface (AWS CLI) pour exécuter une automatisation à l'aide du mode d'exécution manuelle. En utilisant le mode d'exécution manuelle, l'automatisation démarre dans un état Waiting (En attente) et s'arrête dans un état Waiting (En attente). Ceci vous permet de contrôler quand l'automatisation se poursuit, ce qui est utile si vous avez besoin de lire le résultat d'une étape avant de continuer.

Le flux de travail Automation s'exécute dans le contexte de l'utilisateur actuel. Cela signifie que vous n'avez pas besoin de configurer d'autorisations IAM supplémentaires tant que vous avez le droit d'utiliser le runbook et les actions qu'il appelle. Si vous disposez des autorisations d'administrateur dans IAM, vous pouvez exécuter cette automatisation.

Exécution d'une automatisation étape par étape (console)

La procédure suivante montre comment utiliser la console Systems Manager 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, sélectionnez Automation (Automatisation), puis Execute automation (Exécuter l'automatisation).
3. Dans la liste Automation document (Document Automation), sélectionnez un runbook. Sélectionnez une ou plusieurs options dans le panneau Document categories (Catégories de documents) pour filtrer les documents SSM en fonction de leur but. Pour afficher un runbook vous appartenant, sélectionnez l'onglet Owned by me (M'appartenant). Pour afficher un runbook partagé avec votre compte, sélectionnez l'onglet Shared with me (Partagé avec moi). Pour afficher tous les runbooks, sélectionnez l'onglet All documents (Tous les documents).

   > **Note**
   >
   > Vous pouvez consulter les informations sur un runbook en sélectionnant 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 :

- Version par défaut lors de l'exécution : sélectionnez cette option si le runbook d'automatisation est mis à jour régulièrement et qu'une nouvelle version par défaut est attribuée.
- Dernière version lors de l'exécution : sélectionnez cette option si le runbook d'automatisation est mis à jour régulièrement et que vous souhaitez exécuter la dernière version mise à jour.
- 1 (Par défaut) : sélectionnez cette option pour exécuter la première version du document, qui est la version par défaut.

5. Choisissez Next (Suivant).
6. Dans la section Execution Mode (Mode d'exécution), sélectionnez 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 Execute (Exécuter).
9. Sélectionnez Execute this step (Exécuter cette étape) lorsque vous êtes prêt à commencer à la première étape de l'automatisation. L'automatisation démarre avec l'étape une et s'interrompt avant d'exécuter toutes les étapes suivantes spécifiées dans le runbook que vous avez choisies à l'étape 3 de cette procédure. Si le runbook possède plusieurs étapes, vous devez sélectionner Execute this step (Exécuter cette étape) pour chaque étape afin que l'automatisation continue. Chaque fois que vous sélectionnez Execute this step (Exécuter cette étape) l'action s'exécute.

   > **Note**
   >
   > La console affiche le statut de l'automatisation. Si l'automatisation ne parvient pas à exécuter une étape, consultez Résolution des problèmes liés à Systems Manager Automation.

10. Une fois que vous avez terminé toutes les étapes spécifiées dans le runbook, sélectionnez Complete and view results (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 explique comment utiliser la AWS CLI (sous Linux, macOS ou Windows) ou les AWS Tools for PowerShell pour exécuter manuellement une automatisation étape par étape.

Pour exécuter une automatisation étape par étape

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

   Pour plus d'informations, consultez la section Installation ou mise à jour de la version la plus récente de l'AWS CLI et Installation d'AWS Tools for PowerShell.

2. Exécutez la commande suivante pour démarrer une automatisation manuelle. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   Linux & macOS

   ```
   aws ssm start-automation-execution \
       --document-name runbook name \
       --mode Interactive \
       --parameters runbook parameters
   ```

   Windows

   ```
   aws ssm start-automation-execution ^
       --document-name runbook name ^
       --mode Interactive ^
       --parameters runbook parameters
   ```

   PowerShell

   ```
   Start-SSMAutomationExecution `
       -DocumentName runbook name `
       -Mode Interactive `
       -Parameter runbook parameters
   ```

   Voici un exemple d'utilisation du runbook `AWS-RestartEC2Instance` pour redémarrer l'instance EC2 spécifiée.

   Linux & macOS

   ```
   aws ssm start-automation-execution \
       --document-name "AWS-RestartEC2Instance" \
       --mode Interactive \
   ```

```
    --parameters "InstanceId=i-02573cafcfEXAMPLE"
```

Windows

```
aws ssm start-automation-execution ^
    --document-name "AWS-RestartEC2Instance" ^
    --mode Interactive ^
    --parameters "InstanceId=i-02573cafcfEXAMPLE"
```

PowerShell

```
Start-SSMAutomationExecution `
    -DocumentName AWS-RestartEC2Instance `
    -Mode Interactive
    -Parameter @{"InstanceId"="i-02573cafcfEXAMPLE"}
```

Le système retourne des informations telles que les suivantes.

Linux & macOS

```
{
    "AutomationExecutionId": "ba9cd881-1b36-4d31-a698-0123456789ab"
}
```

Windows

```
{
    "AutomationExecutionId": "ba9cd881-1b36-4d31-a698-0123456789ab"
}
```

PowerShell

```
ba9cd881-1b36-4d31-a698-0123456789ab
```

3. Exécutez la commande suivante lorsque vous êtes prêt à démarrer la première étape de l'automatisation. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations. L'automatisation démarre avec l'étape une et s'interrompt avant d'exécuter toutes les étapes suivantes spécifiées dans le runbook que vous

avez choisies à l'étape 1 de cette procédure. Si le runbook comporte plusieurs étapes, vous devez exécuter la commande suivante pour chaque étape pour que l'automatisation continue.

Linux & macOS

```
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 ba9cd881-1b36-4d31-a698-0123456789ab `
    -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 & macOS

```
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 ba9cd881-1b36-4d31-a698-0123456789ab
```

Le système retourne des informations telles que les suivantes.

Linux & macOS

```
{
    "StepExecutions": [
        {
            "StepName": "stopInstances",
            "Action": "aws:changeInstanceState",
            "ExecutionStartTime": 1557167178.42,
            "ExecutionEndTime": 1557167220.617,
            "StepStatus": "Success",
            "Inputs": {
                "DesiredState": "\"stopped\"",
                "InstanceIds": "[\"i-02573cafcfEXAMPLE\"]"
            },
            "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-02573cafcfEXAMPLE\"]"
```

```
            },
            "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-02573cafcfEXAMPLE\"]"
            },
            "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-02573cafcfEXAMPLE\"]"
            },
            "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-02573cafcfEXAMPLE"]]}
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 que toutes les étapes spécifiées dans le runbook choisi sont terminées. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

Linux & macOS

```
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 ba9cd881-1b36-4d31-a698-0123456789ab `
    -Type Complete
```

Il n'y a pas de sortie si la commande réussit.

Planification des automatisations

Les rubriques suivantes contiennent des informations sur la manière de planifier l'exécution des automatisations à un intervalle ou à une heure spécifique que vous indiquez.

Table des matières

Exécution des automatisations avec les associations State Manager

Vous pouvez démarrer une automatisation en créant une association State Manager avec un runbook. State Manager est une fonctionnalité de AWS Systems Manager. Vous pouvez cibler différents types de ressources AWS en créant une association State Manager avec un runbook. Par exemple, vous pouvez créer des associations qui appliquent un état souhaité sur une ressource AWS, y compris les éléments suivants :

- Attachez un rôle Systems Manager à des instances Amazon Elastic Compute Cloud (Amazon EC2) pour en faire des Instances gérées.
- Appliquer les règles d'entrée et de sortie souhaitées pour un groupe de sécurité.
- Créez ou supprimez des sauvegardes Amazon DynamoDB.
- Créez ou supprimez des instantanés Amazon Elastic Block Store (Amazon EBS).
- Désactivez les autorisations de lecture et d'écriture sur des compartiments Amazon Simple Storage Service (Amazon S3).
- Démarrez, redémarrez ou arrêtez des instances gérées et des instances Amazon Relational Database Service (Amazon RDS).
- Appliquez des correctifs à des AMIs Linux, macOS et Windows.

Utilisez les procédures suivantes pour créer une association State Manager qui exécute une automatisation à l'aide de la console AWS Systems Manager et de l'AWS Command Line Interface (AWS CLI).

Avant de commencer

Tenez compte des informations importantes suivantes avant d'exécuter des automatisations à l'aide de State Manager :

- Avant de pouvoir créer une association qui exécute un runbook, vérifiez que vous avez configuré les autorisations pour Automation, une fonctionnalité de AWS Systems Manager. Pour de plus amples informations, veuillez consulter Configuration d'Automation.
- Les associations State Manager qui exécutent des runbooks contribuent au nombre maximal d'automatisations exécutées simultanément dans votre Compte AWS. Vous pouvez avoir un maximum de 100 automatisations simultanées en cours d'exécution à la fois. Pour plus d'informations, veuillez consulter la rubrique Quotas de service Systems Manager dans la Référence générale d'Amazon Web Services.
- Lors de l'exécution d'une automatisation, State Manager ne journalise pas les opérations d'API initiées par l'automatisation dans AWS CloudTrail.
- Systems Manager crée automatiquement un rôle lié au service afin que State Manager ait l'autorisation d'appeler des opérations d'API 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 AWS Tools for PowerShell.

Linux & macOS

```
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 aux services pour Systems Manager.

Création d'une association qui exécute une automatisation (console)

La procédure suivante décrit comment utiliser la console Systems Manager pour créer une association State Manager qui exécute une automatisation.

Pour créer une association State Manager 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, sélectionnez 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, sélectionnez un runbook. Utilisez la barre de recherche pour filtrer les runbooks Document type : Equal : Automation (Type de document : égal : Automation). Pour afficher plus de runbooks, utilisez les nombres à droite de la barre de recherche.

> **Note**
>
> Vous pouvez consulter les informations sur un runbook en sélectionnant son nom.

5. Sélectionnez Simple execution (Exécution simple) pour exécuter l'instance d'Automation sur une ou plusieurs cibles en spécifiant les ID de ressource pour ces cibles. Sélectionnez 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 AWS Resource Groups. Vous pouvez également contrôler l'opération de l'automatisation 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), sélectionnez 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 runbook 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), sélectionnez une méthode pour cibler les ressources.

      - Resource Group (Groupe de ressources) : sélectionnez le nom du groupe dans la liste Resource Group (Groupe de ressources). Pour de plus amples informations sur le ciblage d'AWS Resource Groups dans les runbooks, consultez Ciblage AWS Resource Groups.
      - Balises : entrez la clé de balise et la valeur de balise (le cas échéant) dans les champs fournis. Choisissez Add (Ajouter). Pour de plus amples informations sur le ciblage des balises dans les runbooks, consultez Ciblage d'une balise.
      - Parameter Values (Valeurs des paramètres) : entrez les valeurs dans la section Paramètres d'entrée. Si vous spécifiez plusieurs valeurs, Systems Manager exécute une automatisation sur chaque valeur spécifiée.

        Par exemple, supposons que le runbook comporte un paramètre InstanceID. Si vous ciblez les valeurs du paramètre InstanceID lorsque vous exécutez l'automatisation, Systems Manager exécute une automatisation enfant pour chaque valeur d'ID d'instance spécifiée. L'automatisation parent se termine lorsque l'automatisation a exécuté chaque instance spécifiée ou si l'exécution 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 runbooks, consultez Ciblage des paramètres de valeur.

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 runbook `AWS-RestartEC2Instance` et que vous choisissez de cibler les instances à l'aide de balises, vous n'avez pas à spécifier ou à choisir d'ID d'instance dans la section Input parameters (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 champ AutomationAssumeRole. State Manager utilise le rôle pour appeler des Services AWS spécifiés dans le runbook et exécuter les associations Automation en votre nom.

8. Dans la section Spécifier le programme, sélectionnez On Schedule (Activé) si vous voulez exécuter l'association à intervalles réguliers. Si vous sélectionnez 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, consultez Expressions cron et rate pour les associations.

   > **Note**
   >
   > Les expressions Rate sont le mécanisme de planification préféré pour les associations State Manager qui exécutent des runbooks. Les expressions Rate permettent plus de flexibilité pour exécuter des associations dans le cas où vous avez atteint le nombre maximal d'automatisations s'exécutant simultanément. Avec un programme 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.

   Sélectionnez No schedule (Aucun programme) si vous souhaitez exécuter l'association à un moment donné.

9. (Facultatif) Dans la section Rate Control (Contrôle du débit), sélectionnez les options Concurrency (Simultanéité) et Error threshold (Seuil d'erreur) pour contrôler le déploiement des automatisations dans vos ressources AWS.

   a. Dans la section Simultanéité, sélectionnez une option :

- Sélectionnez targets (cibles) pour entrer un nombre absolu de cibles pouvant exécuter l'automatisation simultanément.
- Sélectionnez percentage (pourcentage) pour indiquer un pourcentage de l'ensemble de cibles pouvant exécuter l'automatisation simultanément.

b. Dans la section Error threshold (Seuil d'erreurs), sélectionnez une option :

- Sélectionnez errors (erreurs) pour indiquer un nombre absolu d'erreurs autorisées avant qu'Automation ne cesse d'envoyer l'automatisation à d'autres ressources.
- Sélectionnez percentage (pourcentage) pour indiquer un pourcentage d'erreurs autorisées avant qu'Automation ne cesse d'envoyer l'automatisation à d'autres ressources.

Pour de plus amples informations sur l'utilisation des cibles et du contrôle du débit avec Automation, consultez Exécution des automatisations à grande échelle.

10. Sélectionnez 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 sélectionnez No schedule (Aucune planification), 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 les AWS Tools for PowerShell pour créer une association State Manager qui exécute une automatisation.

Avant de commencer

Avant d'effectuer la procédure suivante, assurez-vous d'avoir créé un rôle de service IAM contenant les autorisations nécessaires au runbook, et configuré une relation de confiance pour Automation, une fonctionnalité de AWS Systems Manager. Pour de plus amples informations, veuillez consulter Tâche 1 : Création d'un rôle de service pour Automation.

Pour créer une association qui exécute une automatisation

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

   Pour plus d'informations, consultez la section Installation ou mise à jour de la version la plus récente de l'AWS CLI et Installation d'AWS Tools for PowerShell.

2. Exécutez la commande suivante pour afficher une liste de documents.

   Linux & macOS

   ```
   aws ssm list-documents
   ```

   Windows

   ```
   aws ssm list-documents
   ```

   PowerShell

   ```
   Get-SSMDocumentList
   ```

   Notez le nom du runbook que vous souhaitez utiliser pour l'association.

3. Exécutez la commande suivante pour afficher les détails relatifs au runbook. Dans la commande ci-après, remplacez *runbook name* (nom du runbook) par vos propres informations.

   Linux & macOS

   ```
   aws ssm describe-document \
   --name runbook 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 runbook 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 runbook 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 association State Manager. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   Ciblage à l'aide de balises

   Linux & macOS

   ```
   aws ssm create-association \
   --association-name association name \
   --targets Key=tag:key name,Values=value \
   --name runbook name \
   --parameters AutomationAssumeRole=arn:aws:iam::123456789012:role/
   RunbookAssumeRole \
   --automation-target-parameter-name target parameter \
   --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 association name ^
--targets Key=tag:key name,Values=value ^
--name runbook name ^
--parameters AutomationAssumeRole=arn:aws:iam::123456789012:role/
RunbookAssumeRole ^
--automation-target-parameter-name target parameter ^
--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:key name"
$Targets.Values = "value"

New-SSMAssociation `
-AssociationName "association name" `
-Target $Targets `
-Name "runbook name" `
-Parameters @{
"AutomationAssumeRole"="arn:aws:iam::123456789012:role/RunbookAssumeRole" } `
-AutomationTargetParameterName "target parameter" `
-ScheduleExpression "cron or rate expression"
```

> **Note**
>
> Si vous créez une association à l'aide des AWS Tools for 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 & macOS

```
aws ssm create-association \
--association-name association name \
--targets Key=ParameterValues,Values=value,value 2,value 3 \
--name runbook name \
--parameters AutomationAssumeRole=arn:aws:iam::123456789012:role/
RunbookAssumeRole \
--automation-target-parameter-name target parameter \
--schedule "cron or rate expression"
```

Windows

```
aws ssm create-association ^
--association-name association name ^
--targets Key=ParameterValues,Values=value,value 2,value 3 ^
--name runbook name ^
--parameters AutomationAssumeRole=arn:aws:iam::123456789012:role/
RunbookAssumeRole ^
--automation-target-parameter-name target parameter ^
--schedule "cron or rate expression"
```

PowerShell

```
$Targets = New-Object Amazon.SimpleSystemsManagement.Model.Target
$Targets.Key = "ParameterValues"
$Targets.Values = "value","value 2","value 3"

New-SSMAssociation `
-AssociationName "association name" `
-Target $Targets `
-Name "runbook name" `
-Parameters @{
"AutomationAssumeRole"="arn:aws:iam::123456789012:role/RunbookAssumeRole"} `
-AutomationTargetParameterName "target parameter" `
-ScheduleExpression "cron or rate expression"
```

Ciblage à l'aide d'AWS Resource Groups

Linux & macOS

```
aws ssm create-association \
--association-name association name \
--targets Key=ResourceGroup,Values=resource group name \
--name runbook name \
--parameters AutomationAssumeRole=arn:aws:iam::123456789012:role/
RunbookAssumeRole \
--automation-target-parameter-name target parameter \
--schedule "cron or rate expression"
```

Windows

```
aws ssm create-association ^
--association-name association name ^
--targets Key=ResourceGroup,Values=resource group name ^
--name runbook name ^
--parameters AutomationAssumeRole=arn:aws:iam::123456789012:role/
RunbookAssumeRole ^
--automation-target-parameter-name target parameter ^
--schedule "cron or rate expression"
```

PowerShell

```
$Targets = New-Object Amazon.SimpleSystemsManagement.Model.Target
$Targets.Key = "ResourceGroup"
$Targets.Values = "resource group name"

New-SSMAssociation `
-AssociationName "association name" `
-Target $Targets `
-Name "runbook name" `
-Parameters @{
"AutomationAssumeRole"="arn:aws:iam::123456789012:role/RunbookAssumeRole"} `
-AutomationTargetParameterName "target parameter" `
-ScheduleExpression "cron or rate expression"
```

Ciblage de comptes et de régions multiples

Linux & macOS

```
aws ssm create-association \
--association-name association name \
--targets Key=ResourceGroup,Values=resource group name \
--name runbook name \
--parameters AutomationAssumeRole=arn:aws:iam::123456789012:role/
RunbookAssumeRole \
--automation-target-parameter-name target parameter \
--schedule "cron or rate expression" \
--target-locations
 Accounts=111122223333,444455556666,444455556666,Regions=region,region
```

Windows

```
aws ssm create-association ^
--association-name association name ^
--targets Key=ResourceGroup,Values=resource group name ^
--name runbook name ^
--parameters AutomationAssumeRole=arn:aws:iam::123456789012:role/
RunbookAssumeRole ^
--automation-target-parameter-name target parameter ^
--schedule "cron or rate expression" ^
--target-locations
 Accounts=111122223333,444455556666,444455556666,Regions=region,region
```

PowerShell

```
$Targets = New-Object Amazon.SimpleSystemsManagement.Model.Target
$Targets.Key = "ResourceGroup"
$Targets.Values = "resource group name"

New-SSMAssociation `
-AssociationName "association name" `
-Target $Targets `
-Name "runbook name" `
-Parameters @{
"AutomationAssumeRole"="arn:aws:iam::123456789012:role/RunbookAssumeRole"} `
-AutomationTargetParameterName "target parameter" `
-ScheduleExpression "cron or rate expression" `
-TargetLocations @{
```

```
    "Accounts"=["111122223333,444455556666,444455556666"],
    "Regions"=["region,region"]
```

La commande renvoie des détails pour la nouvelle association similaires à ce qui suit.

Linux & macOS

```
{
"AssociationDescription": {
    "ScheduleExpression": "cron(0 7 ? * MON *)",
    "Name": "AWS-StartEC2Instance",
    "Parameters": {
        "AutomationAssumeRole": [
            "arn:aws:iam::123456789012:role/RunbookAssumeRole"
        ]
    },
    "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/RunbookAssumeRole"
        ]
    },
    "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.

Résolution des problèmes liés à l'exécution des automatisations par des associations State Manager

Systems Manager Automation impose une limite de 100 automatisations simultanées et 1 000 automatisations en file d'attente par compte et par région. Si une association State Manager qui exécute un runbook indique le statut Échec et un statut détaillé AutomationExecutionLimitExceeded, votre automatisation pourrait avoir atteint la limite. Par conséquent, Systems Manager limite les automatisations. 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 dotées d'un statut Pending (En suspens). En supprimant ces automatisations, vous désactivez la file d'attente actuelle.

Planifier des automatisations avec des fenêtres de maintenance

Vous pouvez démarrer une automatisation en configurant un runbook en tant que tâche enregistrée pour une fenêtre de maintenance. Si le runbook est enregistré 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 runbook nommé `CreateAMI` qui crée une Amazon Machine Image (AMI) d'instances enregistrées comme cibles dans la fenêtre de maintenance. Pour spécifier le runbook `CreateAMI` (ainsi que l'automatisation correspondante) en tant que tâche enregistrée d'une fenêtre de maintenance, vous devez d'abord créer une fenêtre de maintenance et enregistrer des cibles. Vous utilisez ensuite la procédure suivante afin de spécifier le document `CreateAMI` 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 l'automatisation et crée une AMI de cibles enregistrées.

Pour plus d'informations sur la création de runbooks Automation, consultez Créer vos propres runbooks. L'automatisation est une capacité de AWS Systems Manager.

Utilisez les procédures suivantes pour configurer une automatisation en tant que tâche enregistrée pour une fenêtre de maintenance à l'aide de la AWS Systems Manager console, AWS Command Line Interface (AWS CLI) ou AWS Tools for Windows PowerShell.

Enregistrement d'une automatisation dans une fenêtre de maintenance (console)

La procédure suivante décrit comment utiliser la console Systems Manager 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, consultez les procédures suivantes.

- Créer une fenêtre de maintenance (console).
- Affecter des cibles à une fenêtre de maintenance (console)

Pour configurer une automatisation en tant que tâche enregistrée pour une fenêtre de maintenance

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation de gauche, sélectionnez Maintenance Windows, puis sélectionnez la fenêtre de maintenance avec laquelle vous souhaitez enregistrer une tâche Automation.
3. Sélectionnez Actions. Ensuite, sélectionnez Register Automation task (Enregistrer la tâche d'automatisation) pour exécuter une automatisation sur les cibles à l'aide d'un runbook.
4. Pour Name (Nom), entrez le nom de la tâche.
5. Pour Description, entrez une description.
6. Pour Document, sélectionnez le runbook qui définit les tâches à exécuter.
7. Pour Document version (Version du document), sélectionnez la version du runbook à utiliser.
8. Pour Task priority (Priorité de tâche), sélectionnez une priorité. 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 Targets (Cibles), si le runbook que vous avez choisi est celui qui exécute les tâches sur les ressources, identifiez les cibles sur lesquelles vous voulez exécuter cette automatisation en spécifiant des balises ou en sélectionnant des instances manuellement.

> **Note**
>
> Si vous voulez transmettre les ressources par l'intermédiaire de paramètres d'entrée au lieu de cibles, il est inutile de spécifier une cible de fenêtre de maintenance.
> Dans la plupart des cas, il est inutile de spécifier explicitement une cible pour une tâche d'automatisation. Par exemple, supposons que vous créiez une tâche de type Automation pour mettre à jour une Amazon Machine Image (AMI) pour Linux à l'aide du runbook `AWS-UpdateLinuxAmi`. Lorsque la tâche s'exécute, l'AMI est mise à jour avec les derniers packages de distribution Linux et les logiciels Amazon disponibles. Ces mises à jour sont déjà installées sur les nouvelles instances créées à partir de l'AMI. Comme l'ID de l'AMI à mettre à jour est spécifié dans les paramètres d'entrée du runbook, il est inutile de spécifier à nouveau une cible dans la tâche de la fenêtre de maintenance.

Pour obtenir des informations sur les tâches de la fenêtre de maintenance qui n'ont pas besoin de cibles, consultez the section called “Enregistrement de tâches de fenêtre de maintenance sans cibles”.

10. (Facultatif) Pour Contrôle du débit :

> **Note**
>
> Si la tâche que vous exécutez ne spécifie pas de cibles, il est inutile de spécifier des contrôles de débit.

- Pour Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage de cibles sur lesquelles exécuter l'automatisation en même temps.

  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 d'automatisations qui peuvent s'exécuter en même temps en spécifiant un pourcentage.

Lorsque la fenêtre de maintenance s'exécute, une nouvelle automatisation est initiée par cible. Il y a une limite de 100 automatisations simultanées par. Compte AWS Si vous spécifiez un taux de simultanéité supérieur à 100, les automatisations simultanées supérieures à 100 sont automatiquement ajoutées à la file d'attente de l'automatisation. Pour plus d'informations, veuillez consulter la rubrique Quotas de service Systems Manager dans la Référence générale d'Amazon Web Services.

- Pour Error threshold (Seuil d'erreurs), 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, puis que Systems Manager arrête l'exécution de l'automatisation, alors la quatrième erreur est reçue. Les cibles traitant toujours l'automatisation pourraient également envoyer des erreurs.

11. Dans la section Input Parameters (Paramètres d'entrée), spécifiez les paramètres du runbook. Pour les runbooks, le système fournit automatiquement certaines des valeurs. Vous pouvez conserver ou remplacer ces valeurs.

> **Important**
>
> Pour les runbooks, vous pouvez éventuellement spécifier un rôle responsable Automation. Si vous ne spécifiez pas de rôle pour ce paramètre, l'automatisation endosse le rôle de service de fenêtre de maintenance que vous avez choisi à l'étape 11. Vous devez donc vous assurer que le rôle de service de fenêtre de maintenance que vous choisissez dispose des autorisations AWS Identity and Access Management (IAM) appropriées pour effectuer les actions définies dans le runbook.
> Par exemple, le rôle lié à un service pour Systems Manager ne dispose pas de l'autorisation IAM `ec2:CreateSnapshot`, ce qui est nécessaire pour exécuter le runbook `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 des autorisations `ec2:CreateSnapshot`. Pour plus d'informations, consultez Configuration d'Automation.

12. Dans la zone IAM service role (Fonction du service),choisissez un rôle pour fournir les autorisations permettant à Systems Manager de lancer l'automatisation.

    Pour créer une fonction du service personnalisée pour les tâches de fenêtre de maintenance, consultez la rubrique Utiliser la console pour configurer les autorisations pour les fenêtres de maintenance.

13. Sélectionnez Enregistrer la tâche d'automatisation.

Enregistrement d'une automatisation dans une fenêtre de maintenance (ligne de commande)

La procédure suivante décrit comment utiliser AWS CLI (sous Linux ou Windows) ou comment AWS Tools for PowerShell 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, consultez les procédures suivantes.

- Étape 1 : Création de la fenêtre de maintenance (AWS CLI).
- Étape 2 : Enregistrement d'un nœud cible auprès de la fenêtre de maintenance (AWS CLI)

Pour configurer une automatisation en tant que tâche enregistrée pour une fenêtre de maintenance

1. Installez et configurez le AWS CLI ou le AWS Tools for PowerShell, si ce n'est pas déjà fait.

   Pour plus d'informations, consultez la section Installation ou mise à jour de la version la plus récente de l' AWS CLI et Installation d' AWS Tools for PowerShell.

2. Créez une commande pour configurer une automatisation en tant que tâche enregistrée pour une fenêtre de maintenance. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   Linux & macOS

```
aws ssm register-task-with-maintenance-window \
--window-id window ID \
--name task name \
--task-arn runbook name \
--targets Key=targets,Values=value \
--service-role-arn IAM role arn \
--task-type AUTOMATION \
--task-invocation-parameters task parameters \
--priority task priority \
--max-concurrency 10% \
--max-errors 5
```

> **Note**
>
> Si vous configurez une automatisation en tant que tâche enregistrée à l'aide du paramètre AWS CLI, utilisez le `--Task-Invocation-Parameters` paramètre pour spécifier les paramètres à transmettre à une tâche lors de son exécution. N'utilisez pas le paramètre `--Task-Parameters`. Le paramètre `--Task-Parameters` est un paramètre hérité.
>
> Pour les tâches de la fenêtre de maintenance qui n'ont pas de cible spécifiée, vous ne pouvez pas fournir de valeurs pour `--max-errors` et `--max-concurrency`. Au lieu de cela, le système insère une valeur d'espace réservé de 1, qui peut être rapportée dans la réponse à des commandes telles que describe-maintenance-window-tasks et get-maintenance-window-task. Ces valeurs n'affectent pas l'exécution de votre tâche et peuvent être ignorées.
>
> Pour obtenir des informations sur les tâches de la fenêtre de maintenance qui n'ont pas besoin de cibles, consultez Enregistrement de tâches de fenêtre de maintenance sans cibles.

Windows

```
aws ssm register-task-with-maintenance-window ^
--window-id window ID ^
--name task name ^
--task-arn runbook name ^
--targets Key=targets,Values=value ^
--service-role-arn IAM role arn ^
--task-type AUTOMATION ^
--task-invocation-parameters task parameters ^
--priority task priority ^
--max-concurrency 10% ^
--max-errors 5
```

> **Note**
>
> Si vous configurez une automatisation en tant que tâche enregistrée à l'aide du paramètre AWS CLI, utilisez le `--task-invocation-parameters` paramètre pour spécifier les paramètres à transmettre à une tâche lors de son exécution. N'utilisez

pas le paramètre `--task-parameters`. Le paramètre `--task-parameters` est un paramètre hérité.
Pour les tâches de la fenêtre de maintenance qui n'ont pas de cible spécifiée, vous ne pouvez pas fournir de valeurs pour `--max-errors` et `--max-concurrency`. Au lieu de cela, le système insère une valeur d'espace réservé de 1, qui peut être rapportée dans la réponse à des commandes telles que describe-maintenance-window-tasks et get-maintenance-window-task. Ces valeurs n'affectent pas l'exécution de votre tâche et peuvent être ignorées.
Pour obtenir des informations sur les tâches de la fenêtre de maintenance qui n'ont pas besoin de cibles, consultez Enregistrement de tâches de fenêtre de maintenance sans cibles.

PowerShell

```
Register-SSMTaskWithMaintenanceWindow `
-WindowId window ID `
-Name "task name" `
-TaskArn "runbook name" `
-Target @{ Key="targets";Values="value" } `
-ServiceRoleArn "IAM role arn" `
-TaskType "AUTOMATION" `
-Automation_Parameter @{ "task parameter"="task parameter value"} `
-Priority task priority `
-MaxConcurrency 10% `
-MaxError 5
```

Note

Si vous configurez une automatisation en tant que tâche enregistrée à l'aide du paramètre AWS Tools for PowerShell, utilisez le `-Automation_Parameter` paramètre pour spécifier les paramètres à transmettre à une tâche lors de son exécution. N'utilisez pas le paramètre `-TaskParameters`. Le paramètre `-TaskParameters` est un paramètre hérité.
Pour les tâches de la fenêtre de maintenance qui n'ont pas de cible spécifiée, vous ne pouvez pas fournir de valeurs pour `-MaxError` et `-MaxConcurrency`. Au lieu de cela, le système insère une valeur d'espace réservé de 1, qui peut être rapportée dans la réponse à des commandes telles que `Get-`

`SSMMaintenanceWindowTaskList` et `Get-SSMMaintenanceWindowTask`. Ces valeurs n'affectent pas l'exécution de votre tâche et peuvent être ignorées.
Pour obtenir des informations sur les tâches de la fenêtre de maintenance qui n'ont pas besoin de cibles, consultez Enregistrement de tâches de fenêtre de maintenance sans cibles.

L'exemple suivant configure une automatisation en tant que tâche enregistrée dans une fenêtre de maintenance avec la priorité 1. Il montre également que les options `--targets`, `--max-errors` et `--max-concurrency` peuvent être omises pour une tâche de fenêtre de maintenance sans cible. L'automatisation utilise le runbook `AWS-StartEC2Instance` et l'automatisation spécifiée endosse le rôle de démarrage des 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 chaque instant. En outre, la tâche enregistrée cesse d'être exécutée sur des instances supplémentaires pour un intervalle particulier si le nombre d'erreurs dépasse 1.

Linux & macOS

```
aws ssm register-task-with-maintenance-window \
--window-id mw-0c50858d01EXAMPLE \
--name StartEC2Instances \
--task-arn AWS-StartEC2Instance \
--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
```

Windows

```
aws ssm register-task-with-maintenance-window ^
--window-id mw-0c50858d01EXAMPLE ^
--name StartEC2Instances ^
--task-arn AWS-StartEC2Instance ^
--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
```

PowerShell

```
Register-SSMTaskWithMaintenanceWindow `
-WindowId mw-0c50858d01EXAMPLE `
-Name "StartEC2" `
-TaskArn "AWS-StartEC2Instance" `
-ServiceRoleArn "arn:aws:iam::123456789012:role/MaintenanceWindowRole" `
-TaskType "AUTOMATION" `
-Automation_Parameter
 @{ "InstanceId"="{{TARGET_ID}}";"AutomationAssumeRole"="arn:aws:iam::123456789012:role/
AutomationAssumeRole" } `
-Priority 1
```

La commande renvoie des détails similaires à ce qui suit pour la nouvelle tâche enregistrée.

Linux & macOS

```
{
"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. Remplacez *maintenance windows ID* (ID des fenêtres de maintenance) par vos propres informations.

Linux & macOS

```
aws ssm describe-maintenance-window-tasks \
--window-id maintenance window ID
```

Windows

```
aws ssm describe-maintenance-window-tasks ^
--window-id maintenance window ID
```

PowerShell

```
Get-SSMMaintenanceWindowTaskList `
-WindowId maintenance window ID
```

Le système retourne des informations telles que les suivantes.

Linux & macOS

```
{
"Tasks": [
    {
        "ServiceRoleArn": "arn:aws:iam::123456789012:role/
MaintenanceWindowRole",
        "MaxErrors": "1",
        "TaskArn": "AWS-StartEC2Instance",
        "MaxConcurrency": "1",
        "WindowTaskId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE",
        "TaskParameters": {},
        "Priority": 1,
        "WindowId": "mw-0c50858d01EXAMPLE",
        "Type": "AUTOMATION",
        "Targets": [
        ],
        "Name": "StartEC2"
    }
]
}
```

Windows

```
{
"Tasks": [
    {
        "ServiceRoleArn": "arn:aws:iam::123456789012:role/
MaintenanceWindowRole",
        "MaxErrors": "1",
        "TaskArn": "AWS-StartEC2Instance",
        "MaxConcurrency": "1",
        "WindowTaskId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE",
        "TaskParameters": {},
        "Priority": 1,
        "WindowId": "mw-0c50858d01EXAMPLE",
        "Type": "AUTOMATION",
        "Targets": [
        ],
        "Name": "StartEC2"
    }
]
}
```

PowerShell

```
Description    :
LoggingInfo    :
MaxConcurrency : 5
MaxErrors      : 1
Name           : StartEC2
Priority       : 1
ServiceRoleArn : arn:aws:iam::123456789012:role/MaintenanceWindowRole
Targets        : {}
TaskArn        : AWS-StartEC2Instance
TaskParameters : {}
Type           : AUTOMATION
WindowId       : mw-0c50858d01EXAMPLE
WindowTaskId   : 4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE
```

Référence sur les actions Systems Manager Automation

Cette référence décrit les actions Automation que vous pouvez spécifier dans un runbook Automation. Automation est une fonctionnalité de AWS Systems Manager. Ces actions ne peuvent pas être utilisées dans d'autres types de documents Systems Manager (SSM). Pour de plus amples informations sur les plug-ins pour d'autres types de documents SSM, veuillez consulter Référence de plug-in de document Command.

Systems Manager Automation exécute les étapes définies dans les runbooks 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 runbook.

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 runbooks, 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 runbook.

> **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 runbooks Automation d'AWS (runbooks `AWS-*`) tels que les runbooks `AWS-ConfigureS3BucketLogging`, `AWS-CreateDynamoDBBackup` et `AWS-RestartEC2Instance`, par exemple. Cette exigence s'applique également à tous les runbooks 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 octroyer des autorisations à d'autres Services AWS en ajoutant une politique IAM en ligne au rôle. Pour de plus amples informations, veuillez consulter (Facultatif) Ajoutez une politique d'automatisation en ligne ou une politique gérée par le client pour invoquer d'autres Services AWS.

Rubriques

- aws:approve - Suspendre une automatisation pour approbation manuelle
- aws:assertAwsResourceProperty - Affirmer un statut de ressource AWS ou un statut d'événement
- aws:branch – exécuter les étapes d'automatisation conditionnelle
- aws:changeInstanceState - Modifier ou affirmer le statut de l'instance
- aws:copyImage - Copier ou chiffrer une Amazon Machine Image
- aws:createImage – supprimer une Amazon Machine Image
- aws:createStack— Crée une AWS CloudFormation pile
- aws:createTags - Créer des balises pour des ressources AWS
- aws:deleteImage - Supprimer une Amazon Machine Image
- aws:deleteStack – supprime une pile AWS CloudFormation
- aws:executeAutomation - Exécuter une autre automatisation
- aws:executeAwsApi— Appelez et exécutez des opérations AWS d'API
- aws:executeScript - Exécuter un script
- aws:executeStateMachine – exécuter une machine d'état AWS Step Functions
- aws:invokeWebhook : appeler une intégration de webhook Automation
- aws:invokeLambdaFunction – appeler une fonction AWS Lambda
- aws:loop : itérer les étapes d'une automatisation
- aws:pause - Suspendre une automatisation
- aws:runCommand - Exécuter une commande sur une instance gérée
- aws:runInstances – lancer une instance Amazon EC2
- aws:sleep - Retarder une automatisation
- aws:updateVariable : met à jour la valeur d'une variable runbook
- aws:waitForAwsResourceProperty - Attendre sur une propriété de ressource AWS
- Variables système Automation

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'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.

description

Informations que vous fournissez pour décrire l'objectif d'un runbook ou d'une étape.

Type : chaîne

Obligatoire : non

name

Identifiant qui doit être unique pour tous les noms d'étape dans le runbook.

Type : chaîne

Modèle autorisé : [a-zA-Z0-9_]+$

Obligatoire : oui

action

Nom de l'action que l'étape doit exécuter. `aws:runCommand` - Exécuter une commande sur une instance gérée est un exemple d'action que vous pouvez spécifier ici. Ce document fournit des informations détaillées sur toutes les actions disponibles.

Type : chaîne

Obligatoire : oui

maxAttempts

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 : entier

Obligatoire : non

timeoutSeconds

Valeur 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 : entier

Obligatoire : non

onFailure

Indique si l'automatisation doit être arrêtée, poursuivie ou changer d'étape en cas d'échec. La valeur par défaut de cette option est abort.

Type : chaîne

Valeurs valides : Annuler | Continuer | étape :*nom_étape*

Obligatoire : non

onCancel

Indique quelle étape l'automatisation doit atteindre si un utilisateur annule l'automatisation. Automation exécute le flux d'annulation pendant un maximum de deux minutes.

Type : chaîne

Valeurs valides : Abort | step:*step_name* (Annuler | étape:nom_étape)

Obligatoire : non

La propriété `onCancel` ne prend pas en charge le déplacement vers les actions suivantes :

- `aws:approve`
- `aws:copyImage`
- `aws:createImage`
- `aws:createStack`
- `aws:createTags`
- `aws:loop`
- `aws:pause`
- `aws:runInstances`
- `aws:sleep`

isEnd

Cette option arrête une exécution d'automatisation à la fin d'une étape spécifique. L'automatisation s'arrête si l'étape a échoué ou réussi. La valeur par défaut est false.

Type : booléen

Valeurs valides : true | false

Obligatoire : non

nextStep

Spécifie l'étape de l'automatisation à traiter immédiatement après la fin d'une étape.

Type : chaîne

Obligatoire : non

isCritical

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 évaluée uniquement si vous la définissez explicitement dans votre étape. Si la propriété `onFailure` est définie sur `Continue` dans une étape, la valeur par défaut est FAUX. Sinon, la valeur par défaut de cette option est true.

Type : booléen

Valeurs valides : true | false

Obligatoire : non

inputs

Propriétés spécifiques à l'action.

Type : carte

Obligatoire : oui

Exemple

```
---
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 runbook."
    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: python3.8
    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
  onCancel: step:startInstance
  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 automatisation pour approbation manuelle

Interrompt temporairement une automatisation jusqu'à ce que les principaux désignés aient approuvé ou rejeté l'action. Une fois le nombre d'approbations requises atteint, l'automatisation reprend. Vous pouvez insérer l'étape d'approbation n'importe où dans la section `mainSteps` de votre runbook.

> **Note**
>
> Cette action ne prend pas en charge les automatisations multicomptes et régionales. Le délai d'attente par défaut pour cette action est de 7 jours (604 800 secondes) et la valeur maximale est de 30 jours (2 592 000 secondes). Vous pouvez limiter ou étendre le délai d'attente en spécifiant le paramètre `timeoutSeconds` pour une étape `aws:approve`. 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 l'automatisation jusqu'à ce qu'un approbateur accepte ou rejette l'automatisation. Après approbation, l'automatisation 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

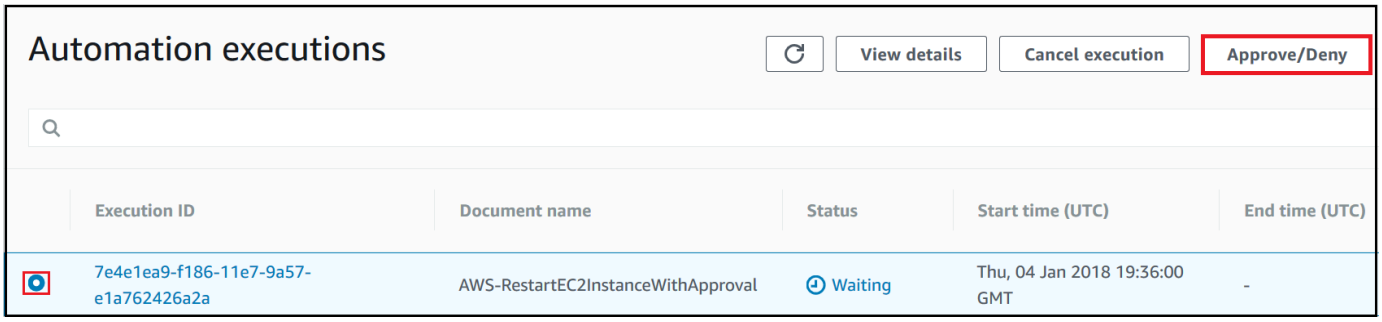
```
{
   "description":"RunInstancesDemo1",
   "schemaVersion":"0.3",
   "assumeRole":"{{ assumeRole }}",
   "parameters":{
      "assumeRole":{
         "type":"String"
      },
      "message":{
         "type":"String"
      }
   },
   "mainSteps":[
      {
```

```
         "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 AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation de gauche, sélectionnez Automation (Automatisation).
3. Sélectionnez l'option en regard d'un flux de travail Automation avec le statut En attente.



4. Sélectionnez Approve/Deny (Approuver/Refuser).
5. Vérifiez les détails de l'automatisation.
6. Sélectionnez Approuver ou Refuser, saisissez un commentaire facultatif, puis sélectionnez Soumettre.

Exemple d'entrée

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

La rubrique Amazon Resource Name (ARN d'un Amazon Simple Notification Service (Amazon SNS) pour les approbations Automation. Lorsque vous spécifiez une étape `aws:approve` dans un runbook, Automation envoie un message à cette rubrique, permettant aux principaux 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 : chaîne

Obligatoire : non

Message

Les informations que vous souhaitez inclure dans la rubrique Amazon SNS lorsque la requête d'approbation est envoyée. Le message peut contenir un nombre maximum de 4 096 caractères.

Type : chaîne

Obligatoire : non

MinRequiredApprovals

Le nombre minimum d'approbations requises pour que l'automatisation 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 : entier

Obligatoire : non

Approbateurs

Liste des principaux AWS authentifiés qui sont en mesure d'approuver ou de rejeter l'action. Le nombre maximum d'approbateurs est de 10. Vous pouvez spécifier des principaux à l'aide des formats suivants :

- Un nom d'utilisateur
- Un ARN d'utilisateur
- Un ARN de rôle IAM

- Un ARN de rôle de responsable IAM

Type : StringList

Obligatoire : oui

EnhancedApprovals

Cette entrée est uniquement utilisée pour les Change Manager modèles. Liste des principaux AWS authentifiés qui peuvent approuver ou rejeter l'action, le type de principal IAM et le nombre minimum d'approbateurs. Voici un exemple :

```
schemaVersion: "0.3"
emergencyChange: false
autoApprovable: false
mainSteps:
    - name: ApproveAction1
    action: aws:approve
    timeoutSeconds: 604800
    inputs:
        Message: Please approve this change request
        MinRequiredApprovals: 3
        EnhancedApprovals:
        Approvers:
            - approver: John Stiles
            type: IamUser
            minRequiredApprovals: 0
            - approver: Ana Carolina Silva
            type: IamUser
            minRequiredApprovals: 0
            - approver: GroupOfThree
            type: IamGroup
            minRequiredApprovals: 0
            - approver: RoleOfTen
            type: IamRole
            minRequiredApprovals: 0
```

Type : StringList

Obligatoire : oui

Sortie

ApprovalStatus

L'état d'approbation de l'étape. L'état peut avoir les valeurs suivantes : Approuvé, Rejeté ou En attente. En attente signifie qu'Automation attend la saisie des approbateurs.

Type : chaîne

ApproverDecisions

Une carte JSON incluant la décision d'approbation de chaque approbateur.

Type : MapList

`aws:assertAwsResourceProperty` - Affirmer un statut de ressource AWS ou un statut 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 attende qu'une instance Amazon Elastic Compute Cloud (Amazon EC2) démarre. Ensuite, l'opération d'API DescribeInstanceStatus d'Amazon EC2 sera appelée, avec la propriété DesiredValue de `running`. Cela garantit que l'automatisation attend qu'une instance soit exécutée, puis qu'elle continue lorsque l'instance est en cours d'exécution.

Pour plus d'exemples sur l'utilisation de cette action, veuillez consulter la rubrique Exemples supplémentaires de runbook.

Entrée

Les entrées sont définies par l'opération d'API que vous sélectionnez.

YAML

```
action: aws:assertAwsResourceProperty
inputs:
  Service: The official namespace of the service
  Api: The API operation or method name
  API operation 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 operation or method name",
    "API operation inputs or parameters":"A value",
    "PropertySelector": "Response object",
    "DesiredValues": [
      "Desired property values"
    ]
  }
}
```

Service

L'espace de noms Service AWS qui contient l'opération 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 voir la liste des espaces de noms Service AWS pris en charge dans la section Available Services (Services disponibles) de la Référence AWS CLI Command.

Type : chaîne

Obligatoire : oui

Api

Le nom de l'opération d'API que vous voulez exécuter. Vous pouvez afficher les opérations 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. Sélectionnez une méthode dans la section Client pour le service que vous voulez appeler. Par exemple, toutes les opérations d'API (méthodes) pour Amazon Relational Database Service (Amazon RDS) sont répertoriées à la page suivante : Méthodes pour Amazon RDS.

Type : chaîne

Obligatoire : oui

Entrées d'opérations d'API

Une ou plusieurs entrées d'opérations 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. Sélectionnez 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. Sélectionnez la méthode describe_db_instances et faites défiler la page 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 operation 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 operation name",
      "API input 1":"A value",
      "API Input 2":"A value",
      "API Input 3":"A value"
}
```

Type : déterminé par l'opération d'API choisie

Obligatoire : oui

PropertySelector

Chemin JSONPath vers un attribut 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. Sélectionnez 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. Sélectionnez la méthode describe_db_instances et faites défiler la page jusqu'à la section Response Structure (Structure d'une réponse). DBInstances est répertorié comme objet de réponse.

Type : chaîne

Obligatoire : oui

DesiredValues

État attendu ou état à partir duquel l'automatisation se poursuit. Si vous spécifiez une valeur booléenne, vous devez utiliser une majuscule, comme True ou False.

Type : StringList

Obligatoire : oui

`aws:branch` – exécuter les étapes d'automatisation conditionnelle

L'action `aws:branch` vous permet de créer une automatisation dynamique qui évalue plusieurs options en une seule étape, puis passe à une autre étape dans le runbook en fonction des résultats de cette évaluation.

Lorsque vous spécifiez l'action `aws:branch` pour une étape, vous définissez des `Choices` que l'automatisation doit évaluer. Les `Choices` peuvent se baser sur une valeur que vous avez spécifiée dans la section `Parameters` du runbook ou sur une valeur dynamique générée comme sortie à partir de l'étape précédente. L'automatisation évalue chaque choix à l'aide d'une expression booléenne. Si le premier choix est true, l'automatisation passe à l'étape désignée pour ce choix. Si le premier choix est false, l'automatisation évalue le choix suivant. L'automatisation continue d'évaluer chaque choix jusqu'à ce qu'il traite un choix défini sur true. L'automatisation accède ensuite à l'étape désignée correspondant au choix défini sur true.

Si aucun des choix n'est true, l'automatisation vérifie si l'étape contient une valeur `default`. La valeur par défaut définit une étape à laquelle l'automatisation doit passer si aucun des choix n'est défini sur true. Si aucune valeur `default` n'est spécifiée pour l'étape, l'automatisation traite l'étape suivante du runbook.

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 runbooks et des exemples utilisant différents opérateurs, veuillez consulter Utilisation d'instructions conditionnelles dans les runbooks.

Entrée

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 runbook 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 si le choix désigné est true, conformément au runbook.

- Variable : spécifiez le nom d'un paramètre qui est défini dans la section `Parameters` du runbook. Vous pouvez également spécifier un objet de sortie d'une étape précédente dans le runbook. 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.
- 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 runbook, le système valide chaque opération dans le runbook. Si une opération n'est pas prise en charge, le système renvoie une erreur lorsque vous tentez de créer le runbook.

Par défaut

Nom d'une étape à laquelle l'automatisation doit passer si aucun des `Choices` n'est true.

Type : chaîne

Obligatoire : non

> **Note**
>
> L'action `aws:branch` prend en charge les opérateurs `And`, `Or` et `Not`. Pour obtenir des exemples `aws:branch` qui utilisent des opérateurs, veuillez consulter Utilisation d'instructions conditionnelles dans les runbooks.

`aws:changeInstanceState` - Modifier ou affirmer le statut de l'instance

Modifie ou affirme l'état de l'instance.

Cette action peut être utilisée en mode déclaration (n'exécute pas l'API pour modifier l'état, mais vérifie que l'instance affiche l'état souhaité.) Pour utiliser le mode assert, définissez le paramètre `CheckStateOnly` 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 Amazon Machine Image (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`.

Entrée

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

ID des instances.

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 : chaîne

Valeurs valides : `running` | `stopped` | `terminated`

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 : chaîne

Obligatoire : non

Sortie

Aucune

`aws:copyImage` - Copier ou chiffrer une Amazon Machine Image

Copie une Amazon Machine Image (AMI) de n'importe quelle Région AWS vers la région actuelle. Cette action peut également chiffrer la nouvelle AMI.

Entrée

Cette action prend en charge la plupart des paramètres `CopyImage`. 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 : chaîne

Obligatoire : oui

SourceImageId

ID d'AMI à copier à partir de la région source.

Type : chaîne

Obligatoire : oui

ImageName

Nom de la nouvelle image.

Type : chaîne

Obligatoire : oui

ImageDescription

Description de l'image cible.

Type : chaîne

Obligatoire : non

Chiffré

Chiffrement de l'AMI cible.

Type : booléen

Obligatoire : non

KmsKeyId

Amazon Resource Name (ARN) complet de la AWS KMS key à utiliser lors du chiffrement des instantanés d'une image pendant une opération de copie. Pour plus d'informations, consultez CopyImage.

Type : 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 : chaîne

Obligatoire : non

Sortie

ImageId

ID de l'image copiée.

ImageState

État de l'image copiée.

Valeurs valides : `available` | `pending` | `failed`

`aws:createImage` – supprimer une Amazon Machine Image

Crée une Amazon Machine Image (AMI) à partir d'une instance en cours d'exécution ou arrêtée.

Entrée

Cette action prend en charge les paramètres `CreateImage` suivants. 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 : chaîne

Obligatoire : oui

ImageName

Nom de l'image.

Type : chaîne

Obligatoire : oui

ImageDescription

Description de l'image.

Type : chaîne

Obligatoire : non

NoReboot

Littéral booléen.

Par défaut, Amazon Elastic Compute Cloud (Amazon EC2) tente d'arrêter et redémarrer l'instance avant de créer l'image. Si l'option No Reboot (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 voulez pas que l'instance s'exécute une fois que vous avez créé une AMI à partir de celle-ci, commencez par utiliser l'action `aws:changeInstanceState` - Modifier ou affirmer le statut de l'instance pour arrêter l'instance, puis utilisez l'action `aws:createImage` avec l'option NoReboot (Pas de redémarrage) définie sur `true`.

Type : booléen

Obligatoire : non

BlockDeviceMappings

Périphériques de stockage en mode bloc pour l'instance.

Type: carte (map)

Obligatoire : non

Sortie

ImageId

ID de l'image nouvellement créée.

Type : chaîne

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.

Type : chaîne

aws:createStack— Crée une AWS CloudFormation pile

Crée une AWS CloudFormation pile à partir d'un modèle.

Pour plus d'informations sur la création de CloudFormation piles, reportez-vous CreateStackà la référence de l'AWS CloudFormation API.

Entrée

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
  Parameters:
    - ParameterKey: LambdaRoleArn
      ParameterValue: "{{LambdaAssumeRole}}"
    - ParameterKey: createdResource
      ParameterValue: createdResource-{{automation:EXECUTION_ID}}
```

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,
        "Parameters": [
          {
            "ParameterKey": "LambdaRoleArn",
            "ParameterValue": "{{LambdaAssumeRole}}"
          },
          {
            "ParameterKey": "createdResource",
            "ParameterValue": "createdResource-{{automation:EXECUTION_ID}}"
          }
    }
}
```

Fonctionnalités

Une liste de valeurs que vous avez spécifiées auparavant CloudFormation peut créer certaines piles. Certains modèles de pile incluent des ressources qui peuvent affecter les autorisations

de votre Compte AWS. 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 capacités. Si vous disposez de ressources IAM avec des noms personnalisés, vous devez spécifier `CAPABILITY_NAMED_IAM`. Si vous ne spécifiez pas ce paramètre, cette action renvoie une erreur `InsufficientCapabilities`. 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 plus d'informations, consultez la section Reconnaissance des ressources IAM dans les AWS CloudFormation modèles.

CAPABILITY_AUTO_EXPAND

Certains modèles contiennent des macros. Les macros effectuent un traitement personnalisé sur les modèles ; cela peut inclure des actions simples telles que des find-and-replace opérations ou des transformations étendues 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 la section Utilisation de AWS CloudFormation macros pour effectuer un traitement personnalisé sur des modèles dans le Guide de AWS CloudFormation l'utilisateur.

Type : tableau de chaînes

Valeurs valides : `CAPABILITY_IAM | CAPABILITY_NAMED_IAM | CAPABILITY_AUTO_EXPAND`

Obligatoire : non

ClientRequestJeton

Identifiant unique pour cette CreateStack demande. Spécifiez ce jeton si vous définissez maxAttempts dans cette étape sur une valeur supérieure à 1. En spécifiant ce jeton CloudFormation , vous savez que vous n'essayez pas de créer une nouvelle pile portant le même nom.

Type : chaîne

Obligatoire : non

Contraintes de longueur : longueur minimum 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 paramètre `DisableRollback` ou le paramètre `OnFailure`, mais pas les deux.

Par défaut : `false`

Type : booléen

Obligatoire : non

NotificationARNs

ARN de rubrique Amazon Simple Notification Service (Amazon SNS) pour la publication d'événements liés aux piles. Vous pouvez trouver les ARN de rubrique SNS en utilisant la console Amazon SNS,https://console.aws.amazon.com/sns/v3/home.

Type : tableau de chaînes

Membres du tableau : Nombre maximum de 5 éléments.

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 paramètre `OnFailure` ou le paramètre `DisableRollback`, mais pas les deux.

Par défaut : `ROLLBACK`

Type : chaîne

Valeurs valides : `DO_NOTHING | ROLLBACK | DELETE`

Obligatoire : non

Paramètres

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. Par exemple : `AWS::EC2::Instance`, `AWS::EC2::*` ou `Custom::MyCustomInstance`. Utilisez la syntaxe suivante pour décrire les types de ressource du modèle.

- Pour toutes les AWS ressources :

```
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 AWS ressource 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, CloudFormation accorde des autorisations à tous les types de ressources. IAM utilise ce paramètre pour les clés de condition CloudFormation spécifiques dans les politiques IAM. Pour plus d'informations, consultez la section Contrôle de l'accès avec AWS Identity and Access Management.

Type : tableau de chaînes

Contraintes de longueur : longueur minimum de 1. Longueur maximale de 256.

Obligatoire : non

RoleARN

Le nom de ressource Amazon (ARN) d'un rôle IAM CloudFormation censé créer la pile. CloudFormation utilise les informations d'identification du rôle pour passer des appels en votre nom. CloudFormation utilise toujours ce rôle pour toutes les futures opérations sur la pile. Tant que les utilisateurs sont autorisés à opérer sur la pile, CloudFormation utilise ce rôle même s'ils n'ont pas l'autorisation de le transmettre. Vérifiez que le rôle accorde le plus faible nombre de privilèges.

Si vous ne spécifiez aucune valeur, CloudFormation utilise le rôle précédemment associé à la pile. Si aucun rôle n'est disponible, CloudFormation utilise une session temporaire générée à partir de vos informations d'identification utilisateur.

Type : 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 : chaîne

Obligatoire : oui

StackPolicyCorps

Structure contenant le corps de la politique de pile. Pour de plus amples informations, veuillez consulter Empêchement des mises à jour des ressources de la pile.

Conditionnel : Vous pouvez spécifier le paramètre `StackPolicyBody` ou le paramètre `StackPolicyURL`, mais pas les deux.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximale de 16384.

Obligatoire : non

StackPolicyURL

Emplacement d'un fichier contenant la politique de pile. L'URL doit pointer vers une politique située dans un compartiment S3 de la même région que la pile. La taille maximum autorisée pour la politique de pile est 16 Ko.

Conditionnel : Vous pouvez spécifier le paramètre `StackPolicyBody` ou le paramètre `StackPolicyURL`, mais pas les deux.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximale de 1350.

Obligatoire : non

Balises

Paires clé-valeur à associer à cette pile. CloudFormation propage également 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 paramètre `TemplateBody` ou le paramètre `TemplateURL`, mais pas les deux.

Type : chaîne

Contraintes de longueur : longueur minimum 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 paramètre `TemplateBody` ou le paramètre `TemplateURL`, mais pas les deux.

Type : chaîne

Contraintes de longueur : Longueur minimum de 1. Longueur maximale de 1024.

Obligatoire : non

TimeoutInMinutes

La durée qui peut s'écouler avant que l'état de la pile ne devienne `CREATE_FAILED`. Si `DisableRollback` n'est pas défini ou est défini sur `false`, la pile sera annulée.

Type : entier

Plage valide : Valeur minimum de 1.

Obligatoire : non

Outputs

StackId

Identifiant unique de la pile.

Type : chaîne

StackStatus

Statut actuel de la pile.

Type : 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

StackStatusMotif

Message d'échec ou de succès associé au statut de la pile.

Type : 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 politique suivante au rôle responsable Automation IAM. Pour de plus amples informations sur le rôle de responsable, veuillez consulter Tâche 1 : Création d'un rôle de service pour Automation.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "sqs:*",
            "cloudformation:CreateStack",
            "cloudformation:DescribeStacks"
         ],
         "Resource":"*"
      }
   ]
}
```

`aws:createTags` - Créer des balises pour des ressources AWS

Crée des balises pour les instances Amazon Elastic Compute Cloud (Amazon EC2) ou les instances gérées AWS Systems Manager.

Entrée

Cette action prend en charge la plupart des paramètres `CreateTags` d'Amazon EC2 et `AddTagsToResource` de Systems Manager. Pour plus d'informations, consultez CreateTags et AddTagsToResource.

L'exemple suivant montre comment baliser une Amazon Machine Image (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

ID des ressources qui doivent être balisées. Si le type de ressources n'est pas « EC2 », ce champ peut uniquement contenir un seul élément.

Type : Liste de chaînes

Obligatoire : oui

Étiquettes

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 : chaîne

Obligatoire : non

Valeurs Valides: `EC2` | `ManagedInstance` | `MaintenanceWindow` | `Parameter`

Sortie

Aucune

`aws:deleteImage` - Supprimer une Amazon Machine Image

Supprime l'Amazon Machine Image (AMI) spécifiée et tous les instantanés associés.

Entrée

Cette action prend en charge un seul paramètre. Pour plus d'informations, consultez la documentation de 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 : chaîne

Obligatoire : oui

Sortie

Aucune

aws:deleteStack – supprime une pile AWS CloudFormation

Supprime une pile AWS CloudFormation.

Entrée

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 CloudFormation sache que vous ne tentez pas de supprimer une pile avec le même nom. Vous pouvez réessayer les demandes `DeleteStack` pour vérifier que CloudFormation les a reçues.

Type : chaîne

Contraintes de longueur : Longueur minimum de 1. Longueur maximum de 128.

Modèle : [a-zA-Z][-a-zA-Z0-9]*

Obligatoire : non

RetainResources.member.N

Cette entrée s'applique uniquement aux piles qui ont un statut `DELETE_FAILED`. Une liste d'ID de ressource logiques pour les ressources vous souhaitez conserver. Pendant la suppression, 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

L'Amazon Resource Name (ARN) d'un rôle AWS Identity and Access Management (IAM) assumé par CloudFormation pour créer la pile. CloudFormation utilise des informations d'identification du rôle pour effectuer des appels en votre nom. CloudFormation utilisera toujours ce rôle pour toutes les opérations futures sur la pile. Tant que les utilisateurs ont l'autorisation d'opérer sur la pile,

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, CloudFormation utilise le rôle qui était précédemment associé à la pile. Si aucun rôle n'est disponible, CloudFormation utilise une session temporaire qui est générée à partir de vos informations d'identification utilisateur.

Type : 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 : chaîne

Obligatoire : oui

Considérations sur la sécurité

Avant de pouvoir utiliser l'action `aws:deleteStack`, vous devez attribuer la politique suivante au rôle responsable Automation IAM. Pour de plus amples informations sur le rôle de responsable, veuillez consulter Tâche 1 : Création d'un rôle de service pour Automation.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "sqs:*",
            "cloudformation:DeleteStack",
            "cloudformation:DescribeStacks"
         ],
         "Resource":"*"
      }
   ]
}
```

`aws:executeAutomation` - Exécuter une autre automatisation

Exécute une automatisation secondaire en appelant un runbook secondaire. Par cette action, vous pouvez créer des runbooks pour vos opérations les plus courantes et vous y référer durant une automatisation. Cette action peut simplifier vos runbooks en supprimant la nécessité de dupliquer les étapes sur les runbooks 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 le même rôle ou utilisateur AWS Identity and Access Management (IAM) que l'utilisateur qui a lancé 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 politique 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, consultez Méthode 2 : Utiliser IAM afin de configurer des rôles pour Automation.

Entrée

YAML

```
name: Secondary_Automation
action: aws:executeAutomation
maxAttempts: 3
timeoutSeconds: 3600
onFailure: Abort
inputs:
  DocumentName: secondaryAutomation
  RuntimeParameters:
    instanceIds:
    - i-1234567890abcdef0
```

JSON

```
{
   "name":"Secondary_Automation",
```

```
   "action":"aws:executeAutomation",
   "maxAttempts":3,
   "timeoutSeconds":3600,
   "onFailure":"Abort",
   "inputs":{
      "DocumentName":"secondaryAutomation",
      "RuntimeParameters":{
         "instanceIds":[
            "i-1234567890abcdef0"
         ]
      }
   }
}
```

DocumentName

Le nom du runbook secondaire à exécuter pendant cette étape. Pour les runbooks dans le même Compte AWS, spécifiez le nom du runbook. Pour les runbooks partagés à partir d'un Compte AWS différent, spécifiez l'Amazon Resource Name (ARN) du runbook. Pour obtenir des informations sur l'utilisation de runbooks partagés, consultez Utilisation de documents SSM partagés.

Type : chaîne

Obligatoire : oui

DocumentVersion

La version du runbook secondaire à exécuter. Si elle n'est pas spécifiée, Automation exécute la version du runbook par défaut.

Type : chaîne

Obligatoire : non

MaxConcurrency

Nombre maximum de cibles pour lesquelles cette tâche peut être exécutée en parallèle. Vous pouvez spécifier un nombre tel que 10 ou un pourcentage tel que 10 %.

Type : chaîne

Obligatoire : non

MaxErrors

Nombre d'erreurs autorisées avant que le système ne cesse d'exécuter l'automatisation sur des cibles supplémentaires. 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 l'exécution de l'automatisation à la réception de la quatrième erreur. Si vous spécifiez 0, le système cesse d'exécuter l'automatisation à des cibles supplémentaires une fois que le premier résultat d'erreur est renvoyé. Si vous exécutez une automatisation sur 50 ressources et que vous définissez `MaxErrors` sur 10 %, le système cesse d'exécuter l'automatisation sur des cibles supplémentaires à réception de la sixième erreur.

Les automatisations qui sont déjà en cours d'exécution quand le seuil `MaxErrors` est atteint sont autorisées à se terminer, mais certaines de ces automatisations peuvent également échouer. Si vous devez vous assurer que le nombre d'échecs d'automatisations ne dépassera pas la valeur `MaxErrors` spécifiée, définissez `MaxConcurrency` sur 1 de sorte que les automatisations s'exécutent une à la fois.

Type : chaîne

Obligatoire : non

RuntimeParameters

Paramètres requis pour le runbook secondaire. Le mappage utilise le format suivant : {"parameter1" : "value1", "parameter2" : "value2" }

Type: carte (map)

Obligatoire : non

Étiquettes

Métadonnées facultatives que vous affectez à une ressource. Vous pouvez spécifier cinq balises maximum pour une automatisation.

Type : MapList

Obligatoire : non

TargetLocations

Un emplacement est une combinaison des Régions AWS et/ou des Comptes AWS où vous voulez exécuter l'automatisation. Vous devez spécifier un nombre minimum de 1 élément et un nombre maximum de 100 éléments.

Type : MapList

Obligatoire : non

TargetMaps

Une liste de mappages clé-valeur des paramètres du document aux ressources cibles. Il n'est pas possible de spécifier `Targets` et `TargetMaps` ensemble.

Type : MapList

Obligatoire : non

TargetParameterName

Le nom du paramètre utilisé comme ressource cible pour l'automatisation à débit contrôlé. Requis uniquement si vous spécifiez `Targets`.

Type : chaîne

Obligatoire : non

Cibles

Une liste de mappages clé-valeur aux ressources cibles. Requis uniquement si vous spécifiez `TargetParameterName`.

Type : MapList

Obligatoire : non

Sortie

Sortie

La sortie générée par l'automatisation secondaire. Vous pouvez référencer la sortie en utilisant le format suivant : *`Secondary_Automation_Step_Name`*.Output

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 de l'automatisation secondaire.

Type : chaîne

État

Le statut de l'automatisation secondaire.

Type : chaîne

`aws:executeAwsApi`— Appelez et exécutez des opérations AWS d'API

Appelle et exécute des opérations d' AWS API. La plupart des opérations d'API sont prises en charge, bien que toutes n'aient pas été testées. Les opérations de l'API de streaming, telles que l'GetObjectopération, ne sont pas prises en charge. Si vous ne savez pas si une opération d'API que vous souhaitez utiliser est une opération de streaming, consultez la documentation Boto3 du service pour déterminer si une API nécessite des entrées ou des sorties de streaming. Nous mettons régulièrement à jour la version Boto3 utilisée par cette action. Cependant, après la sortie d'une nouvelle version de Boto3, quelques semaines peuvent être nécessaires pour que les modifications soient prises en compte dans cette action. Chaque action `aws:executeAwsApi` peut être exécutée jusqu'à une durée maximale de 25 secondes. Pour plus d'exemples sur l'utilisation de cette action, veuillez consulter la rubrique Exemples supplémentaires de runbook.

Inputs

Les entrées sont définies par l'opération d'API que vous sélectionnez.

YAML

```
action: aws:executeAwsApi
inputs:
  Service: The official namespace of the service
  Api: The API operation or method name
  API operation 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 operation or method name",
      "API operation 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 Service AWS noms qui contient l'opération d'API que vous souhaitez exécuter. Vous pouvez consulter la liste des Service AWS espaces de noms pris en charge dans les services disponibles du AWS SDK for 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 Elastic Compute Cloud (Amazon EC2) est `ec2`.

Type : chaîne

Obligatoire : oui

Api

Le nom de l'opération d'API que vous voulez exécuter. Vous pouvez afficher les opérations 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. Sélectionnez une méthode dans la section Client pour le service que vous voulez appeler. Par exemple, toutes les opérations d'API (méthodes) pour Amazon Relational Database Service (Amazon RDS) sont répertoriées à la page suivante : Méthodes pour Amazon RDS.

Type : chaîne

Obligatoire : oui

Entrées d'opérations d'API

Une ou plusieurs entrées d'opérations 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. Sélectionnez 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 vers le bas pour voir les paramètres disponibles, tels que DB InstanceIdentifier, Name et Values.

YAML

```
inputs:
  Service: The official namespace of the service
  Api: The API operation 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 operation name",
      "API input 1":"A value",
      "API Input 2":"A value",
      "API Input 3":"A value"
}
```

Type : déterminé par l'opération d'API choisie

Obligatoire : oui

Outputs

Les sorties sont spécifiées par l'utilisateur en fonction de la réponse de l'opération d'API choisie.

Nom

Nom de la sortie.

Type : chaîne

Obligatoire : oui

Selector

Chemin JSONPath vers un attribut 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. Sélectionnez 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. Sélectionnez la méthode describe_db_instances et faites défiler la page jusqu'à la section Response Structure (Structure d'une réponse). DBInstances est répertorié comme objet de réponse.

Type : entier, booléen StringList, StringMap chaîne ou MapList

Obligatoire : oui

Type

Type de données de l'élément de réponse.

Type : variable

Obligatoire : oui

`aws:executeScript` - Exécuter un script

Exécute le Python ou le PowerShell script fourni à l'aide du runtime et du gestionnaire spécifiés. Chaque action `aws:executeScript` peut être exécutée jusqu'à une durée maximale de 600 secondes (10 minutes). Vous pouvez limiter le délai d'attente en spécifiant le paramètre `timeoutSeconds` pour une étape `aws:executeScript`.

Utilisez les instructions de retour dans votre fonction pour ajouter des sorties à votre charge utile de sortie. Pour des exemples de définition de sorties pour votre action `aws:executeScript`, consultez Exemple 2 : runbook scripté. Vous pouvez également envoyer le résultat des `aws:executeScript` actions de vos runbooks vers le groupe de CloudWatch journaux Amazon Logs que vous spécifiez. Pour plus d'informations, consultez Journalisation de la sortie d'actions Automation avec CloudWatch Logs.

Si vous souhaitez envoyer le résultat des `aws:executeScript` actions vers CloudWatch Logs, ou si les scripts que vous spécifiez pour les `aws:executeScript` actions appellent des opérations d' AWS API, un rôle de service AWS Identity and Access Management (IAM) (ou un rôle assumé) est toujours requis pour exécuter le runbook.

L'`aws:executeScript`action contient les modules PowerShell Core préinstallés suivants :

- Microsoft. PowerShell.Hôte
- Microsoft. PowerShell. Gestion

- Microsoft. PowerShell.Sécurité
- Microsoft. PowerShell.Utilitaire
- PackageManagement
- PowerShellGet

Pour utiliser des modules PowerShell Core qui ne sont pas préinstallés, votre script doit installer le module avec l'`-Force`indicateur, comme indiqué dans la commande suivante. Le module `AWSPowerShell.NetCore` n'est pas pris en charge. *`ModuleName`*Remplacez-le par le module que vous souhaitez installer.

```
Install-Module ModuleName -Force
```

Pour utiliser les applets de commande PowerShell Core dans votre script, nous vous recommandons d'utiliser les `AWS.Tools` modules, comme indiqué dans les commandes suivantes. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations.

- cmdlets Amazon S3.

  ```
  Install-Module AWS.Tools.S3 -Force
  Get-S3Bucket -BucketName bucketname
  ```

- cmdlets Amazon EC2.

  ```
  Install-Module AWS.Tools.EC2 -Force
  Get-EC2InstanceStatus -InstanceId instanceId
  ```

- Applets de AWS Tools for Windows PowerShell commande 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 de commande PowerShell Core, vous devez également importer le module comme indiqué dans la commande suivante.

```
Install-Module AWS.Tools.EC2 -Force
Import-Module AWS.Tools.EC2
```

```
$tag = New-Object Amazon.EC2.Model.Tag
$tag.Key = "Tag"
$tag.Value = "TagValue"

New-EC2Tag -Resource i-02573cafcfEXAMPLE -Tag $tag
```

Pour des exemples d'installation et d'importation de `AWS.Tools` modules, ainsi que d'utilisation d'applets de commande PowerShell Core dans des runbooks, consultez. Créer des runbooks à l'aide de Document Builder

Entrée

Fournissez les informations requises pour exécuter votre script. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

> **Note**
>
> La pièce jointe d'un script Python peut être un fichier .py ou .zip contenant le script. PowerShell les scripts doivent être stockés dans des fichiers .zip.

YAML

```
action: "aws:executeScript"
inputs:
 Runtime: runtime
 Handler: "functionName"
 InputPayload:
  scriptInput: '{{parameterValue}}'
 Script: |-
   def functionName(events, context):
   ...
 Attachment: "scriptAttachment.zip"
```

JSON

```
{
    "action": "aws:executeScript",
    "inputs": {
```

```
        "Runtime": "runtime",
        "Handler": "functionName",
        "InputPayload": {
            "scriptInput": "{{parameterValue}}"
        },
        "Attachment": "scriptAttachment.zip"
    }
}
```

Environnement d'exécution

Langage d'exécution à utiliser pour exécuter le script fourni. `aws:executeScript`prend en charge les scripts Python 3.7 (python3.7), Python 3.8 (python3.8), Python 3.9 (python3.9) Python 3.10 (python3.10), Python 3.11 (python3.11) Core 6.0 (dotnetcore2.1) et 7.0 (dotnetcore3.1 PowerShell). PowerShell

Valeurs prises en charge : **`python3.7` `python3.8`** | **`python3.9`** | **`python3.10`** | | **`python3.11`** | **`PowerShell Core 6.0`** | **`PowerShell 7.0`**

Type : chaîne

Obligatoire : oui

Handler (Gestionnaire)

Le nom de votre fonction. Vous devez vous assurer que la fonction définie dans le gestionnaire possède deux paramètres, `events` et `context`. Le PowerShell moteur d'exécution ne prend pas en charge ce paramètre.

Type : chaîne

Obligatoire : Oui (Python) | Non pris en charge (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 : chaîne

Obligatoire : non

Python

```
description: Tag an instance
schemaVersion: '0.3'
assumeRole: '{{AutomationAssumeRole}}'
parameters:
    AutomationAssumeRole:
        type: String
        description: '(Required) The Amazon Resource Name (ARN) of the IAM role
 that allows Automation to perform the actions on your behalf. If no role is
 specified, Systems Manager Automation uses your IAM permissions to operate this
 runbook.'
    InstanceId:
        type: String
        description: (Required) The ID of the EC2 instance you want to tag.
mainSteps:
  - name: tagInstance
    action: 'aws:executeScript'
    inputs:
        Runtime: "python3.8"
        Handler: tagInstance
        InputPayload:
            instanceId: '{{InstanceId}}'
        Script: |-
          def tagInstance(events,context):
            import boto3

            #Initialize client
            ec2 = boto3.client('ec2')
            instanceId = events['instanceId']
            tag = {
                "Key": "Env",
                "Value": "Example"
            }
            ec2.create_tags(
                Resources=[instanceId],
                Tags=[tag]
            )
```

PowerShell

```
description: Tag an instance
schemaVersion: '0.3'
```

```
assumeRole: '{{AutomationAssumeRole}}'
parameters:
    AutomationAssumeRole:
        type: String
        description: '(Required) The Amazon Resource Name (ARN) of the IAM role
 that allows Automation to perform the actions on your behalf. If no role is
 specified, Systems Manager Automation uses your IAM permissions to operate this
 runbook.'
    InstanceId:
        type: String
        description: (Required) The ID of the EC2 instance you want to tag.
mainSteps:
  - name: tagInstance
    action: 'aws:executeScript'
    inputs:
        Runtime: PowerShell 7.0
        InputPayload:
          instanceId: '{{InstanceId}}'
        Script: |-
          Install-Module AWS.Tools.EC2 -Force
          Import-Module AWS.Tools.EC2

          $input = $env:InputPayload | ConvertFrom-Json

          $tag = New-Object Amazon.EC2.Model.Tag
          $tag.Key = "Env"
          $tag.Value = "Example"

          New-EC2Tag -Resource $input.instanceId -Tag $tag
```

Script

Script incorporé que vous souhaitez exécuter pendant l'automatisation.

Type : chaîne

Obligatoire : Non (Python) | Oui (PowerShell)

Réseau de transit par passerelle

Nom d'un fichier de script autonome ou .zip pouvant être appelé par l'action. Indiquez la même valeur que le `Name` du fichier de pièce jointe du document que vous indiquez dans le paramètre de requête `Attachments`. Pour plus d'informations, consultez Pièces jointes dans AWS Systems Manager la référence API . Le fournissement d'un script à l'aide d'une pièce jointe nécessite

également la définition d'une section `files` dans les éléments de niveau supérieur de votre runbook. Pour plus d'informations, consultez Version de schéma 0.3.

Pour appeler un fichier pour Python, utilisez le format `filename.method_name` dans `Handler`.

> **Note**
>
> La pièce jointe d'un script Python peut être un fichier .py ou .zip contenant le script. PowerShell les scripts doivent être stockés dans des fichiers .zip.

Lorsque vous incluez des bibliothèques Python dans votre pièce jointe, nous vous recommandons d'ajouter un fichier `__init__.py` vide dans chaque répertoire de module. Cela vous permet d'importer les modules de la bibliothèque de votre pièce jointe dans le contenu de votre script. Par exemple : `from library import module`

Type : chaîne

Obligatoire : non

Sortie

Charge utile

Représentation JSON de l'objet renvoyé par votre fonction. Jusqu'à 100 Ko sont renvoyés. La génération d'une liste permet le renvoi d'un maximum de 100 éléments.

`aws:executeStateMachine` – exécuter une machine d'état AWS Step Functions

Exécute une machine d'état AWS Step Functions.

Entrée

Cette action prend en charge la plupart des paramètres de l'opération d'API StartExecution.

Autorisations AWS Identity and Access Management (IAM) nécessaires

- `states:DescribeExecution`
- `states:StartExecution`

- `states:StopExecution`

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

L'Amazon Resource Name (ARN) de la machine d'état Step Functions.

Type : chaîne

Obligatoire : oui

name

Nom de l'exécution.

Type : chaîne

Obligatoire : non

input

Chaîne qui contient les données d'entrée JSON pour l'exécution.

Type : chaîne

Obligatoire : non

Outputs

Les sorties suivantes sont prédéfinies pour cette action.

executionArn

ARN de l'exécution.

Type : chaîne

input

Chaîne qui contient les données d'entrée JSON de l'exécution. Les contraintes de longueur s'appliquent à la taille de la charge utile et sont exprimées en octets dans l'encodage UTF-8.

Type : chaîne

name

Nom de l'exécution.

Type : chaîne

output

Données de sortie JSON de l'exécution. Les contraintes de longueur s'appliquent à la taille de la charge utile et sont exprimées en octets dans l'encodage UTF-8.

Type : chaîne

startDate

Date de début de l'exécution.

Type : chaîne

stateMachineArn

ARN de la machine d'état exécutée.

Type : chaîne

status

Statut actuel de l'exécution.

Type : chaîne

stopDate

Si l'exécution est déjà terminée, date à laquelle l'exécution s'est arrêtée.

Type : chaîne

`aws:invokeWebhook` : appeler une intégration de webhook Automation

Appelle l'intégration du webhook Automation spécifiée. Pour plus d'informations sur la création d'intégrations Automation, consultez Création d'intégrations webhook pour Automation.

> **Note**
>
> Pour exécuter l'action `aws:invokeWebhook`, votre rôle d'utilisateur ou de service doit autoriser les actions suivantes :
>
> - ssm:GetParameter
> - kms:Decrypt
>
> L'autorisation pour l'opération `Decrypt` de AWS Key Management Service (AWS KMS) n'est nécessaire que si vous utilisez une clé gérée par le client pour chiffrer le paramètre de votre intégration.

Entrée

Fournissez les informations relatives à l'intégration d'Automation que vous souhaitez appeler.

YAML

```
action: "aws:invokeWebhook"
inputs:
 IntegrationName: "exampleIntegration"
 Body: "Request body"
```

JSON

```
{
    "action": "aws:invokeWebhook",
    "inputs": {
        "IntegrationName": "exampleIntegration",
        "Body": "Request body"
    }
}
```

IntegrationName

Nom de l'intégration d'Automation. Par exemple, `exampleIntegration`. L'intégration que vous spécifiez doit déjà exister.

Type : chaîne

Obligatoire : oui

Corps de texte

La charge utile que vous souhaitez envoyer lorsque votre intégration de webhook est appelée.

Type : chaîne

Obligatoire : non

Sortie

Réponse

Texte reçu de la réponse du fournisseur de webhook.

ResponseCode

Code d'état HTTP reçu de la réponse du fournisseur de webhook.

`aws:invokeLambdaFunction` – appeler une fonction AWS Lambda

Appelle la fonction AWS Lambda spécifiée.

> **Note**
>
> Chaque action `aws:invokeLambdaFunction` peut être exécutée jusqu'à une durée maximale de 300 secondes (5 minutes). Vous pouvez limiter le délai d'attente en spécifiant le paramètre `timeoutSeconds` pour une étape `aws:invokeLambdaFunction`.

Entrée

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

Nom de la fonction Lambda. Cette fonction doit exister.

Type : chaîne

Obligatoire : oui

Qualificateur

Nom de version ou d'alias de la fonction.

Type : chaîne

Obligatoire : non

InvocationType

Type d'appel. La valeur par défaut est `RequestResponse`.

Type : chaîne

Valeurs valides : `Event` | `RequestResponse` | `DryRun`

Obligatoire : non

LogType

Si la valeur par défaut est `Tail`, le type d'appel doit être `RequestResponse`. Lambda retourne les 4 derniers Ko de données de journalisation générés par votre fonction Lambda, codés en base64.

Type : chaîne

Valeurs valides : `None` | `Tail`

Obligatoire : non

ClientContext

Informations spécifiques au client.

Obligatoire : non

InputPayload

Objet JSON ou YAML transmis au premier paramètre du gestionnaire. Utilisez cette saisie pour transmettre des données vers la fonction. Cette entrée offre plus de flexibilité et de prise en charge que l'entrée `Payload` héritée. Si vous définissez les deux `InputPayload` et `Payload` pour l'action, `InputPayload` prend la priorité et `Payload` la valeur n'est pas utilisée.

Type : StringMap

Obligatoire : non

Charge utile

Objet JSON ou YAML transmis au premier paramètre du gestionnaire. Utilisez cette saisie pour la transmission des données vers la fonction. Nous vous recommandons d'utiliser `InputPayload` entrée pour des fonctionnalités supplémentaires.

Type : chaîne

Obligatoire : non

Sortie

StatusCode

Codes d'état HTTP.

FunctionError

Si cette valeur est présente, elle indique qu'une erreur s'est produite lors de l'exécution de la fonction. Les détails sur l'erreur sont inclus dans la charge utile de la réponse.

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`. Jusqu'à 200 Ko sont renvoyés

Ce qui suit est une partie du runbook `AWS-PatchInstanceWithRollback` démontrant comment référencer les sorties de l'action `aws:invokeLambdaFunction`.

YAML

```
- name: IdentifyRootVolume
  action: aws:invokeLambdaFunction
```

```
  inputs:
    FunctionName: "IdentifyRootVolumeLambda-{{automation:EXECUTION_ID}}"
    Payload: '{"InstanceId": "{{InstanceId}}"}'
- name: PrePatchSnapshot
  action: aws:executeAutomation
  inputs:
    DocumentName: "AWS-CreateSnapshot"
    RuntimeParameters:
      VolumeId: "{{IdentifyRootVolume.Payload}}"
      Description: "ApplyPatchBaseline restoration case contingency"
```

JSON

```
{
    "name": "IdentifyRootVolume",
    "action": "aws:invokeLambdaFunction",
    "inputs": {
      "FunctionName": "IdentifyRootVolumeLambda-{{automation:EXECUTION_ID}}",
      "Payload": "{\"InstanceId\": \"{{InstanceId}}\"}"
    }
  },
  {
    "name": "PrePatchSnapshot",
    "action": "aws:executeAutomation",
    "inputs": {
      "DocumentName": "AWS-CreateSnapshot",
      "RuntimeParameters": {
        "VolumeId": "{{IdentifyRootVolume.Payload}}",
        "Description": "ApplyPatchBaseline restoration case contingency"
      }
    }
  }
```

aws:loop : itérer les étapes d'une automatisation

Cette action itère sur un sous-ensemble d'étapes dans un runbook d'Automatisation. Vous pouvez sélectionner un style boucle do `while` ou `for each`. Pour construire une boucle do `while`, utilisez le paramètre d'entrée `LoopCondition`. Pour construire une boucle `for each`, utilisez les paramètres d'entrée `Iterators` et `IteratorDataType`. Lorsque vous utilisez une action `aws:loop`, spécifiez uniquement le paramètre d'entrée `Iterators` ou `LoopCondition`. Le nombre maximal d'itérations est de 100.

La propriété onCancel ne peut être définie que pour les étapes définies dans une boucle. La propriété onCancel n'est pas prise en charge pour l'action aws:loop.

Exemples

Les exemples suivants montrent comment construire les différents types d'actions de boucle.

do while

```
name: RepeatMyLambdaFunctionUntilOutputIsReturned
action: aws:loop
inputs:
    Steps:
    - name: invokeMyLambda
        action: aws:invokeLambdaFunction
        inputs:
        FunctionName: LambdaFunctionName
        outputs:
        - Name: ShouldRetry
            Selector: $.Retry
            Type: Boolean
    LoopCondition:
        Variable: "{{ invokeMyLambda.ShouldRetry }}"
        BooleanEquals: true
    MaxIterations: 3
```

for each

```
name: stopAllInstancesWithWaitTime
action: aws:loop
inputs:
    Iterators: "{{ DescribeInstancesStep.InstanceIds }}"
    IteratorDataType: "String"
    Steps:
    - name: stopOneInstance
        action: aws:changeInstanceState
        inputs:
        InstanceIds:
            - "{{stopAllInstancesWithWaitTime.CurrentIteratorValue}}"
        CheckStateOnly: false
        DesiredState: stopped
    - name: wait10Seconds
        action: aws:sleep
```

```
        inputs:
        Duration: PT10S
```

Entrée

L'entrée est comme suit.

Itérateurs

La liste des éléments sur lesquels les étapes doivent être itérées. Le nombre maximal d'itérateurs est de 100.

Type : StringList

Obligatoire : non

IteratorDataType

Un paramètre facultatif permettant de spécifier le type de données du `Iterators`. Une valeur pour ce paramètre peut être fournie en même temps que le paramètre d'entrée `Iterators`. Si vous ne spécifiez aucune valeur pour ce paramètre et `Iterators`, vous devez alors spécifier une valeur pour le paramètre `LoopCondition`.

Type : chaîne

Valeurs valides : booléen | entier | chaîne | StringMap

Par défaut : String

Obligatoire : non

LoopCondition

Comprend une `Variable` et une condition de l'opérateur à évaluer. Si vous ne spécifiez aucune valeur pour ce paramètre, vous devez alors spécifier des valeurs pour les paramètres `Iterators` et `IteratorDataType`. Vous pouvez utiliser des évaluations d'opérateurs complexes en combinant les opérateurs `And`, `Not` et `Or`. La condition est évaluée une fois les étapes de la boucle terminées. Si la condition est `true` et que la valeur `MaxIterations` n'a pas été atteinte, les étapes de la boucle sont à nouveau exécutées. Les conditions de l'opérateur sont les suivantes :

Opérations de chaîne

- StringEquals

- EqualsIgnoreÉtui
- StartsWith
- EndsWith
- Contains

Opérations numériques

- NumericEquals
- NumericGreater
- NumericLesser
- NumericGreaterOrEquals
- NumericLesser
- NumericLesserOrEquals

Opération booléenne

- BooleanEquals

Type : StringMap

Obligatoire : non

MaxIterations

Le nombre maximal de fois où les étapes de la boucle sont exécutées. Une fois que la valeur spécifiée pour cette entrée est atteinte, la boucle cesse de fonctionner même si `LoopCondition` est encore `true` ou s'il reste des objets dans le paramètre `Iterators`.

Type : entier

Valeurs valides : de 1 à 100

Obligatoire : non

Étapes

Liste des étapes à exécuter dans la boucle. Elles fonctionnent comme un runbook imbriqué. Au cours de ces étapes, vous pouvez accéder à la valeur actuelle de l'itérateur pour une boucle `for each` à l'aide de la syntaxe `{{loopStepName.CurrentIteratorValue}}`. Vous pouvez également accéder à une valeur entière de l'itération en cours pour les deux types de boucle à l'aide de la syntaxe `{{loopStepName.CurrentIteration}}`.

Type : liste des étapes

Obligatoire : oui

Sortie

CurrentIteration

L'itération de boucle en cours sous forme d'entier. Valeur des itérations commençant à 1.

Type : entier

CurrentIteratorValeur

Valeur de l'itérateur actuel sous forme de chaîne. Cette sortie n'est présente que dans les boucles `for each`.

Type : chaîne

`aws:pause` - Suspendre une automatisation

Cette action suspend l'automatisation. Une fois l'automatisation interrompue, son statut est Waiting (En attente). Pour poursuivre l'automatisation, utilisez l'opération d'API SendAutomationSignal avec le type de signal `Resume`. Nous vous recommandons d'utiliser l'action `aws:sleep` ou `aws:approve` pour effectuer un contrôle plus précis de vos flux de travail.

Entrée

L'entrée est comme suit.

YAML

```
name: pauseThis
action: aws:pause
inputs: {}
```

JSON

```
{
    "name": "pauseThis",
    "action": "aws:pause",
```

```
    "inputs": {}
}
```

Sortie

Aucune

`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 AWS Systems Manager Run Command. Un runbook peut inclure plusieurs actions Run Command, mais la sortie est prise en charge pour une seule action à la fois.

Entrée

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

Si le document de type Commande vous appartient AWS, ou spécifiez le nom du document. Si vous utilisez un document partagé avec un autre Compte AWS, spécifiez l'Amazon Resource Name (ARN) du document. Pour plus d'informations sur l'utilisation de documents partagés, consultez Utilisation de documents SSM partagés.

Type : chaîne

Obligatoire : oui

InstanceIds

ID des instances dans lesquelles vous souhaitez que la commande s'exécute. Vous pouvez spécifier un maximum de 50 ID.

Vous pouvez également utiliser le pseudo-paramètre `{{RESOURCE_ID}}` à la place des ID d'instance pour exécuter la commande sur toutes les instances du groupe cible. Pour plus d'informations sur les pseudo-paramètres, consultez Utilisation de pseudo-paramètres lors de l'enregistrement des tâches de la fenêtre de maintenance.

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 Elastic Compute Cloud (Amazon EC2). Pour de plus amples informations sur l'utilisation du paramètre `Targets`, veuillez consulter Exécuter des commandes à grande échelle.

Type : StringList

Obligatoire : Non (Si vous ne spécifiez pas InstanceIds ou n'utilisez pas le `{{RESOURCE_ID}}` pseudo paramètre, vous devez le `Targets` spécifier.)

Cibles

Ensemble de critères de recherche qui cible les instances en utilisant une combinaison clé/valeur que vous spécifiez. `Targets` est obligatoire si vous ne fournissez pas un ou plusieurs ID d'instance dans l'appel. Pour de plus amples informations sur l'utilisation du paramètre `Targets`, veuillez consulter Exécuter des commandes à grande échelle.

Type : MapList (Le schéma de la carte dans la liste doit correspondre à l'objet.) Pour de plus amples informations, veuillez consulter Cible dans la Référence d'API AWS Systems Manager .

Obligatoire : Non (Si vous ne le spécifiez pas`Targets`, 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"
            ]
        }
    }
}
```

Paramètres

Paramètres obligatoires et facultatifs spécifiés dans le document.

Type: carte (map)

Obligatoire : non

CloudWatchOutputConfig

Options de configuration pour envoyer une sortie de commande à Amazon CloudWatch Logs. Pour plus d'informations sur l'envoi d'une sortie de commande à CloudWatch Logs, consultezConfiguration d'Amazon CloudWatch Logs pour Run Command.

Type : StringMap (Le schéma de la carte doit correspondre à l'objet. Pour plus d'informations, consultez CloudWatchOutputConfigla référence de AWS Systems Manager l'API).

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
        }
    }
}
```

Comment

Informations définies par l'utilisateur au sujet de la commande.

Type : chaîne

Obligatoire : non

DocumentHash

Hachage pour le document.

Type : chaîne

Obligatoire : non

DocumentHashType

Type de hachage.

Type : chaîne

Valeurs valides : `Sha256` | `Sha1`

Obligatoire : non

NotificationConfig

Configurations d'envoi des notifications.

Obligatoire : non

Sorties 3 BucketName

Nom du compartiment S3 pour les réponses de sortie de la commande.

Type : chaîne

Obligatoire : non

Sorties 3 KeyPrefix

Préfixe.

Type : chaîne

Obligatoire : non

ServiceRoleArn

L'ARN du rôle AWS Identity and Access Management (IAM).

Type : chaîne

Obligatoire : non

TimeoutSeconds

Temps d'attente, en secondes, avant qu'une commande ne soit délivrée AWS Systems Manager SSM Agent à une instance. Si la commande n'est pas reçue par l'SSM Agent sur l'instance avant expiration de la valeur spécifiée, le statut de la commande devient `Delivery Timed Out`.

Type : entier

Obligatoire : non

Valeurs valides : 30-2592000

Sortie

CommandId

ID de la commande.

Statut

Statut de la commande.

ResponseCode

Code de réponse de la commande. Si le document que vous exécutez comporte plus d'une étape, aucune valeur n'est renvoyée pour cette sortie.

Sortie

Sortie de la commande. Si vous ciblez une balise ou plusieurs instances avec votre commande, aucune valeur de sortie n'est renvoyée. Vous pouvez utiliser les opérations `GetCommandInvocation` et `ListCommandInvocations` API pour récupérer la sortie pour des instances individuelles.

`aws:runInstances` – lancer une instance Amazon EC2

Lance une nouvelle instance Amazon Elastic Compute Cloud (Amazon EC2).

Entrée

L'action prend en charge la plupart des paramètres de l'API. Pour plus d'informations, consultez la documentation de l'API RunInstances.

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"
             }
          ]
       }
    ]
 }
}
```

AdditionalInfo

Instances réservées.

Type : 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 : chaîne

Obligatoire : non

DisableApiTermination

Active ou désactive la résiliation de l'API d'instance.

Type : booléen

Obligatoire : non

EbsOptimized

Active ou désactive l'optimisation Amazon Elastic Block Store (Amazon EBS).

Type : booléen

Obligatoire : non

IamInstanceProfileArn

Amazon Resource Name (ARN) du profil d'instance AWS Identity and Access Management (IAM) à associer à l'instance.

Type : chaîne

Obligatoire : non

IamInstanceProfileName

Nom du profil d'instance IAM pour l'instance.

Type : chaîne

Obligatoire : non

ImageId

ID de l'Amazon Machine Image (AMI).

Type : chaîne

Obligatoire : oui

InstanceInitiatedShutdownBehavior

Indique si l'instance s'arrête ou est résiliée lors de l'arrêt du système.

Type : chaîne

Obligatoire : non

InstanceType

Type d'instance.

> **Note**
>
> Si aucune valeur de type d'instance n'est fournie, le type d'instance m1.small est utilisé.

Type : chaîne

Obligatoire : non

KernelId

ID du noyau.

Type : chaîne

Obligatoire : non

KeyName

Nom de la paire de clés.

Type : chaîne

Obligatoire : non

MaxInstanceCount

Nombre maximum d'instances à lancer.

Type : chaîne

Obligatoire : non

MetadataOptions

Options de métadonnées de l'instance. Pour plus d'informations, veuillez consulter la rubrique InstanceMetadataOptionsRequest.

Type : StringMap

Obligatoire : non

MinInstanceCount

Nombre minimum d'instances à lancer.

Type : chaîne

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 IPv4 principale.

Type : chaîne

Obligatoire : non

RamdiskId

ID du disque RAM.

Type : chaîne

Obligatoire : non

SecurityGroupIds

ID des groupes de sécurité de l'instance.

Type : StringList

Obligatoire : non

SecurityGroups

Noms des groupes de sécurité de l'instance.

Type : StringList

Obligatoire : non

SubnetId

ID de sous-réseau.

Type : 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 des ressources AWS.

Type : MapList (pour plus d'informations, consultez TagSpecification)

Obligatoire : non

UserData

Script fourni en tant que valeur de littéral de chaîne. Si une valeur littérale est entrée, elle doit être codée en base64.

Type : chaîne

Obligatoire : non

Sortie

InstanceIds

ID des instances.

InstanceStates

Statut actuel de l'instance.

`aws:sleep` - Retarder une automatisation

Retarde 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.

Entrée

Vous pouvez retarder une automatisation 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 une automatisation 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**
>
> L'automatisation prend en charge un délai maximal de 604 799 secondes (7 jours).

Durée

Une durée ISO 8601. Vous ne pouvez pas spécifier une durée négative.

Type : 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 : chaîne

Obligatoire : non

Sortie

Aucune

`aws:updateVariable` : met à jour la valeur d'une variable runbook

Cette action met à jour la valeur d'une variable runbook. Le type de données de la valeur doit correspondre au type de données de la variable que vous voulez mettre à jour. Les conversions par type de données ne sont pas prises en charge. La propriété `onCancel` n'est pas prise en charge pour l'action `aws:updateVariable`.

Entrée

L'entrée est comme suit.

YAML

```
name: updateStringList
action: aws:updateVariable
inputs:
    Name: variable:variable name
    Value:
    - "1"
    - "2"
```

JSON

```
{
    "name": "updateStringList",
    "action": "aws:updateVariable",
    "inputs": {
        "Name": "variable:variable name",
        "Value": ["1","2"]
    }
}
```

Nom

Nom de la variable dont vous voulez mettre à jour la valeur. Vous devez utiliser le format `variable:variable name`

Type : chaîne

Obligatoire : oui

Valeur

La nouvelle valeur à attribuer à la variable. La valeur doit correspondre au type de données de la variable. Les conversions par type de données ne sont pas prises en charge.

Type : Booléen | Entier | | Chaîne MapList | | StringList StringMap

Obligatoire : oui

Contraintes :

- MapList peut contenir un maximum de 200 éléments.
- La clé peut avoir une longueur minimale de 1 et une longueur maximale de 50.
- StringList peut être un nombre minimum de 0 éléments et un nombre maximum de 50 éléments.
- Les chaînes peuvent avoir une longueur minimale de 1 et une longueur maximale de 512.

Sortie

Aucun

`aws:waitForAwsResourceProperty` - Attendre sur une propriété de ressource AWS

L'action `aws:waitForAwsResourceProperty` permet à votre automatisation d'attendre un statut de ressource ou d'événement spécifique avant de se poursuivre. Pour plus d'exemples sur l'utilisation de cette action, veuillez consulter la rubrique Exemples supplémentaires de runbook.

> **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:waitForAwsResourceProperty`. Pour de plus amples informations et des exemples sur l'utilisation de cette action, veuillez consulter Gestion de délais d'expiration dans des runbooks.

Entrée

Les entrées sont définies par l'opération d'API que vous sélectionnez.

YAML

```
action: aws:waitForAwsResourceProperty
inputs:
  Service: The official namespace of the service
  Api: The API operation or method name
  API operation 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 operation or method name",
    "API operation inputs or parameters":"A value",
    "PropertySelector": "Response object",
    "DesiredValues": [
      "Desired property value"
    ]
  }
}
```

Service

L'espace de noms Service AWS qui contient l'opération d'API que vous souhaitez exécuter. Par exemple, l'espace de noms pour AWS Systems Manager est `ssm`. L'espace de noms pour Amazon Elastic Compute Cloud (Amazon EC2) est `ec2`. Vous pouvez voir la liste des espaces de noms Service AWS pris en charge dans la section Available Services (Services disponibles) de la Référence AWS CLI Command.

Type : chaîne

Obligatoire : oui

Api

Le nom de l'opération d'API que vous voulez exécuter. Vous pouvez afficher les opérations 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. Sélectionnez une méthode dans la section Client pour le service que vous voulez appeler. Par exemple, toutes les opérations d'API (méthodes) pour Amazon Relational Database Service (Amazon RDS) sont répertoriées à la page suivante : Méthodes pour Amazon RDS.

Type : chaîne

Obligatoire : oui

Entrées d'opérations d'API

Une ou plusieurs entrées d'opérations 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. Sélectionnez 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. Sélectionnez la méthode describe_db_instances et faites défiler la page 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 operation 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 operation name",
      "API input 1":"A value",
      "API Input 2":"A value",
      "API Input 3":"A value"
}
```

Type : déterminé par l'opération d'API choisie

Obligatoire : oui

PropertySelector

Chemin JSONPath vers un attribut 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. Sélectionnez 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. Sélectionnez la méthode describe_db_instances et faites défiler la page jusqu'à la section Response Structure (Structure d'une réponse). DBInstances est répertorié comme objet de réponse.

Type : chaîne

Obligatoire : oui

DesiredValues

État attendu ou état à partir duquel l'automatisation se poursuit.

Type : MapList, StringList

Obligatoire : oui

Variables système Automation

Les runbooks Automation AWS Systems Manager utilisent les variables suivantes. Pour obtenir un exemple d'utilisation de ces variables, affichez la source JSON du runbook `AWS-UpdateWindowsAmi`.

Pour afficher la source JSON du runbook **`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 runbook **`AWS-UpdateWindowsAmi`**.
4. Sélectionnez l'onglet Contenu.

Variables système

Les runbooks Automation prennent en charge actuellement les variables système suivantes.

Variable	Détails
`global:ACCOUNT_ID`	L'ID de l'Compte AWS de l'utilisateur ou du rôle dans lequel Automation s'exécute.
`global:DATE`	Date (au moment de l'exécution) au format AAAA-MM-JJ.

Variable	Détails
`global:DATE_TIME`	Date et heure (au moment de l'exécution) au format AAAA-MM-JJ_HH.mm.ss.
`global:AWS_PARTITION`	Partition dans laquelle se trouve la ressource. Pour les Régions AWS standards, la partition est `aws`. Pour les ressources dans d'autres partitions, la partition est `aws-`*`partition name`* . Par exemple, la partition des ressources dans la région AWS GovCloud (US-West) est `aws-us-gov` .
`global:REGION`	La région dans laquelle le runbook est exécuté. Par exemple, us-east-2.

Variables d'automatisation

Les runbooks Automation prennent en charge les variables d'automatisation suivantes.

Variable	Détails
`automation:EXECUTION_ID`	Identifiant unique attribué à l'exécution de l'automatisation actuelle. Par exemple, `1a2b3c-1a2b3c-1a2b3c-1a2b3c 1a2b3c1a2b3c` .

Rubriques

Terminologie

Les conditions suivantes décrivent comment les variables et les paramètres sont résolus.

Durée	Définition	Exemple
ARN constant	Un Amazon Resource Name (ARN) valide, sans variables.	`arn:aws:iam::123456789012:role/roleName`
Paramètre runbook	Un paramètre défini au niveau du runbook (`instanceId` , par exemple). 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 runbook lors de l'évaluation d'une partie du runbook.	`"activities": [ { "id": "copyImage", "activityType": "AWS-CopyImage", "maxAttempts": 1, "onFailure": "Continue", "inputs": { "ImageName": "{{imageName}}", "SourceImageId": "{{sourceImageId}}", "SourceRegion": "{{sourceRegion}}",`

Durée	Définition	Exemple
		"Encrypted": true, "ImageDescription": "Test CopyImage Description created on **{{global:DATE}}** " } }]

Durée	Définition	Exemple
Variable d'automatisation	Variable relative à l'automatisation remplacée dans le runbook lors de l'évaluation d'une partie du runbook.	<pre>{ "name": "runFixedCmds", "action": "aws:runCommand", "maxAttempts": 1, "onFailure": "Continue", "inputs": { "DocumentName": "AWS-RunPowerShellScript", "InstanceIds": ["{{LaunchInstance.InstanceIds}}"], "Parameters": { "commands": ["dir", "date", "“{{outputFormat}}” -f “left”,”right”,”{{global:DATE}}”,” **{{automation:EXECUTION_ID}}** ”] } } }</pre>

Durée	Définition	Exemple
Paramètre Systems Manager	Une variable définie dans AWS Systems Manager Parameter Store. Il ne peut pas être référencé directeme nt 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: '{{Automa tionAssumeRole}}' parameters: AutomationAssumeRo le: 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 runbook. LatestAmi: type: String default: >- {{ssm:/aws/ service/ami-wind ows-latest/Windows _Server-2016-English- Full-Base}} description: The latest Windows Server 2016 AMI queried from the public parameter. mainSteps: - name: launchIns tance action: 'aws:runI nstances' maxAttempts: 3`

Durée	Définition	Exemple
		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 l'utilisateur appelant est autorisé à passer le `assumeRole` donné.	{ "description": "Test all Automation resolvable parameters", "schemaVersion": "0.3", "assumeRole": ***"arn:aws:iam::123456789012:role/roleName"*** , "parameters": { ...
Paramètre runbook fourni pour `AssumeRole` au démarrage de l'automatisation.	Doit être défini dans la liste des paramètres du runbook.	{ "description": "Test all Automation resolvable parameters", "schemaVersion": "0.3", "assumeRole": **"{{dynamicARN}}"** , "parameters": { ...

Scénario	Commentaires	Exemple
Valeur fournie pour le paramètre du runbook au démarrage.	Le client fournit la valeur à utiliser pour un paramètre. Toutes les entrées fournies au moment du démarrage doivent être définies dans la liste des paramètres du runbook.	<pre>... "parameters": { "amiId": { "type": "String", "default": "ami-12345678 ", "description": "list of commands to run as part of first step" }, ...</pre> Les entrées pour démarrer l'exécution d'Automation incluent : `{"amiId" : ["ami-12345678 "] }`

Scénario	Commentaires	Exemple
Paramètre Systems Manager référencé dans le contenu du runbook.	La variable existe dans le compte du client, ou est un paramètre accessibl e publiquement, et le `AssumeRole` pour le runbook 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-wind ows-latest/Windows _Server-2016-English- Full-Base}} description: The latest Windows Server 2016 AMI queried from the public parameter. mainSteps: - name: launchIns tance action: 'aws:runI nstances' maxAttempts: 3 timeoutSeconds: 1200 onFailure: Abort inputs: ImageId: '{{Latest Ami}}' ... ```

Scénario	Commentaires	Exemple
Variable système référencée dans la définition de l'étape	Au démarrage de l'automatisation, une variable système est remplacée dans le runbook. La valeur injectée dans le runbook 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 runbook.	<pre>... "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}}}"] } } }, ...</pre>

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 runbook. 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}}" }" } } ...

Scénario	Commentaires	Exemple
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 de l'étape précédente est référencé e à l'aide de la syntaxe `{{stepName.OutputName}}` . Cette syntaxe ne peut pas être utilisée par le client pour les paramètres du runbook. Ceci est résolu lorsque l'étape de référence s'exécute. Le paramètre n'est pas répertorié dans les paramètres du runbook.	<pre>... "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" } } ...</pre>

Scénarios non pris en charge

Scénario	Comment	Exemple
Paramètre Systems Manager fourni pour `assumeRole` au moment de la création	Non pris en charge.	`...` `{` `  "description":` `"Test all Automation` `resolvable parameter` `s",` `  "schemaVersion":` `"0.3",` `  "assumeRole":` `"{{ssm:administrato` `rRoleARN}}  ",` `  "parameters": {` `...`
Paramètre Systems Manager directement référencé dans l'entrée de l'étape.	Renvoie une exception `InvalidDocumentCon tent` au moment de la création.	`...` `mainSteps:` `  - name: launchIns` `tance` `    action: 'aws:runI` `nstances'` `    maxAttempts: 3` `    timeoutSeconds:` `1200` `    onFailure: Abort` `    inputs:` `      ImageId: '{{ssm:/` `aws/service/ami-win` `dows-latest/Window` `s_Server-2016-Engl` `ish-Full-Base}}'` `...`

Scénario	Comment	Exemple
Définition de l'étape des variables	La définition d'une étape du runbook 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	Comment	Exemple
Paramètres runbook de références croisées	L'utilisateur fournit un paramètre d'entrée au moment du démarrage qui est une référence à un autre paramètre du runbook.	... "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	Comment	Exemple
Extension à plusieurs niveaux	Le runbook 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}}*** }}"] }

Scénario	Comment	Exemple
		`` ... Note: The customer intention here would be to run a command of "echo {Hello world}" ```

Scénario	Comment	Exemple
Référencement de sortie à partir d'une étape de runbook qui est un autre type de variable	L'utilisateur fait référence à la sortie d'une étape d'un précédent runbook 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:execu teAwsApi 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.Image IdList}} 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:copyI mage requires a "String" type variable but the preceding step ```

Scénario	Comment	Exemple
		`outputs a "StringList" type variable.`

Créer vos propres runbooks

Un manuel d'automatisation définit les actions que Systems Manager exécute sur vos instances gérées et sur d'autres AWS ressources lorsqu'une automatisation s'exécute. L'automatisation est une capacité de AWS Systems Manager. Un runbook 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 automation (automatisation).

Les types d'actions pris en charge pour les runbooks vous permettent d'automatiser une grande variété d'opérations dans votre AWS environnement. Par exemple, en utilisant le type `executeScript` d'action, vous pouvez intégrer un python ou un PowerShell script directement dans votre runbook. (Lorsque vous créez un runbook 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 `deleteStack` d'action `createStack` et. En outre, en utilisant le type `executeAwsApi` d'action, une étape peut exécuter n'importe quelle opération d'API Service AWS, y compris la création ou la suppression de AWS ressources, le démarrage d'autres processus, le lancement de notifications, etc.

Pour obtenir la liste des 20 types d'action pris en charge pour l'automatisation, consultez Référence sur les actions Systems Manager Automation.

AWS Systems Manager L'automatisation propose plusieurs runbooks avec des étapes prédéfinies que vous pouvez utiliser pour effectuer des tâches courantes, telles que le redémarrage d'une ou plusieurs instances Amazon Elastic Compute Cloud (Amazon EC2) ou la création d'une (). Amazon Machine Image AMI Vous pouvez également créer vos propres runbooks et les partager avec d'autres utilisateurs Comptes AWS, ou les rendre publics pour tous les utilisateurs d'Automation.

Les runbooks sont écrits en utilisant YAML ou JSON. Toutefois, l'outil Document Builder de la console Systems Manager Automation permet de créer un runbook sans avoir à utiliser 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 runbooks Automation d' AWS (runbooks `AWS-*`) tels que les runbooks `AWS-ConfigureS3BucketLogging`, `AWS-CreateDynamoDBBackup` et `AWS-RestartEC2Instance`, par exemple. Cette exigence s'applique également à tous les runbooks d'automatisation personnalisés que vous créez et qui invoquent d'autres services Services AWS en utilisant des 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 accorder des autorisations à d'autres personnes en Services AWS ajoutant une politique IAM intégrée au rôle. Pour plus d'informations, consultez (Facultatif) Ajoutez une politique d'automatisation en ligne ou une politique gérée par le client pour invoquer d'autres Services AWS.

Pour plus d'informations concernant les actions que vous pouvez spécifier dans un runbook, consultez Référence sur les actions Systems Manager Automation.

Pour plus d'informations sur l'utilisation du AWS Toolkit for Visual Studio Code pour créer des runbooks, consultez les documents Working with Systems Manager Automation dans le guide de l'AWS Toolkit for Visual Studio Code utilisateur.

Pour plus d'informations sur l'utilisation du concepteur visuel pour créer un runbook personnalisé, consultezExpérience de conception visuelle pour les runbook d'automatisation.

Table des matières

Expérience de conception visuelle pour les runbook d'automatisation

Automation AWS Systems Manager fournit une expérience de conception visuelle à faible code qui vous aide à créer des runbook d'automatisation. L'expérience de conception visuelle fournit une interface glisser-déposer avec la possibilité d'ajouter votre propre code afin que vous puissiez créer

et modifier des runbook plus facilement. L'expérience de conception visuelle vous permet d'effectuer les actions suivantes :

- Contrôlez les instructions conditionnelles.
- Contrôlez la manière dont les entrées et les sorties sont filtrées ou transformées pour chaque action.
- Configurez la gestion des erreurs.
- Prototypez de nouveaux runbook.
- Utilisez vos prototypes de runbook comme point de départ pour le développement local avec le AWS Toolkit for Visual Studio Code.

Lorsque vous créez ou modifiez un runbook, vous pouvez accéder à l'expérience de conception visuelle depuis la console Automation. Lorsque vous créez un runbook, l'expérience de conception visuelle valide votre travail et génère automatiquement du code. Vous pouvez consulter le code généré ou l'exporter pour le développement local. Lorsque vous avez terminé, vous pouvez enregistrer votre runbook, l'exécuter et examiner les résultats dans la console Systems Manager Automation.

Avant de commencer

Pour utiliser l'expérience de conception visuelle, vous avez besoin d'un identifiant Compte AWS et d'informations d'identification fournissant les autorisations appropriées pour toutes les ressources que vous souhaitez utiliser.

Dans l'expérience de conception visuelle, l'automatisation s'intègre à la sécurité Amazon CodeGuru pour vous aider à détecter les violations de la politique de sécurité et les vulnérabilités dans vos scripts Python. Pour utiliser cette fonctionnalité pour les actions `aws:executeScript`, votre politique (IAM) AWS Identity and Access Management doit inclure les autorisations suivantes :

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "codeguru-security:CreateUploadUrl",
                "codeguru-security:CreateScan",
                "codeguru-security:GetScan",
```

```
                "codeguru-security:GetFindings"
            ]
        }
    ]
}
```

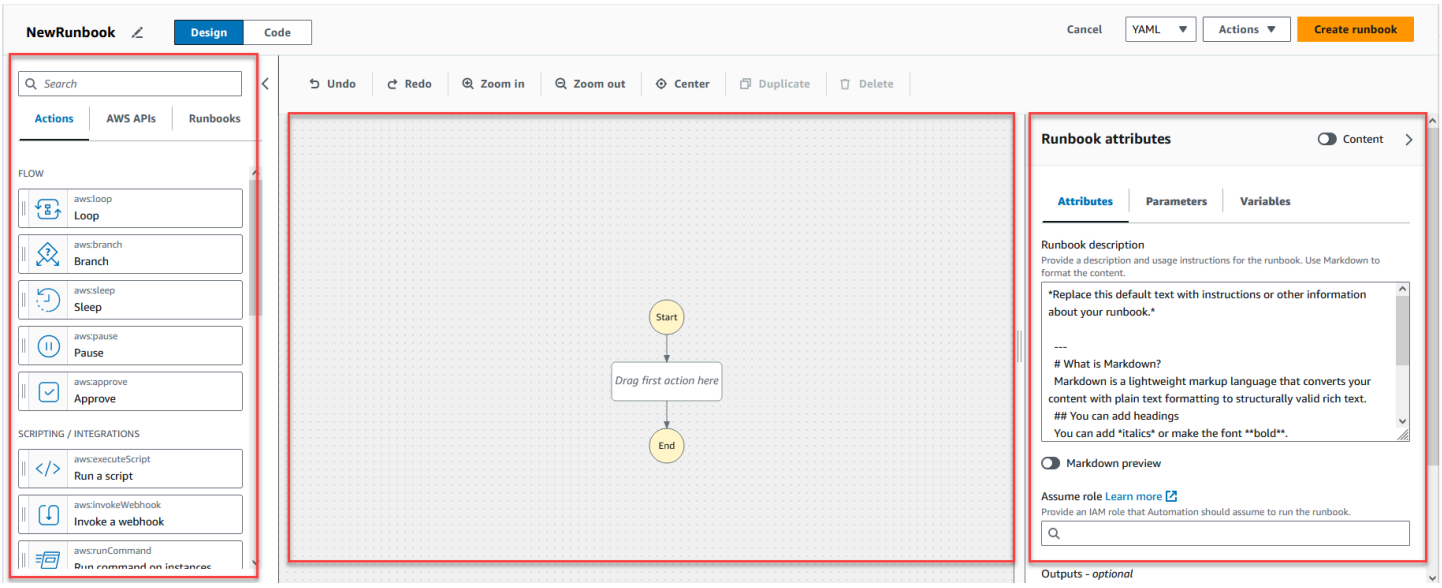
Rubriques

- Présentation de l'interface d'expérience de conception visuelle
- Utilisation de l'expérience de conception visuelle
- Configurer les entrées et les sorties pour vos actions
- Gestion des erreurs grâce à l'expérience de conception visuelle
- Tutoriel : Création d'un runbook à l'aide de l'expérience de conception visuelle

Présentation de l'interface d'expérience de conception visuelle

L'expérience de conception visuelle de Systems Manager Automation est un concepteur de flux de travail visuel à faible code qui vous aide à créer des runbook d'automatisation.

Découvrez l'expérience de conception visuelle grâce à un aperçu des composants de l'interface :



- Le navigateur Actions contient les onglets Actions, API AWS et Runbook.
- Le canevas vous permet de glisser-déposer des actions dans votre graphe de flux de travail, de modifier l'ordre des actions et de sélectionner les actions à configurer ou à afficher.

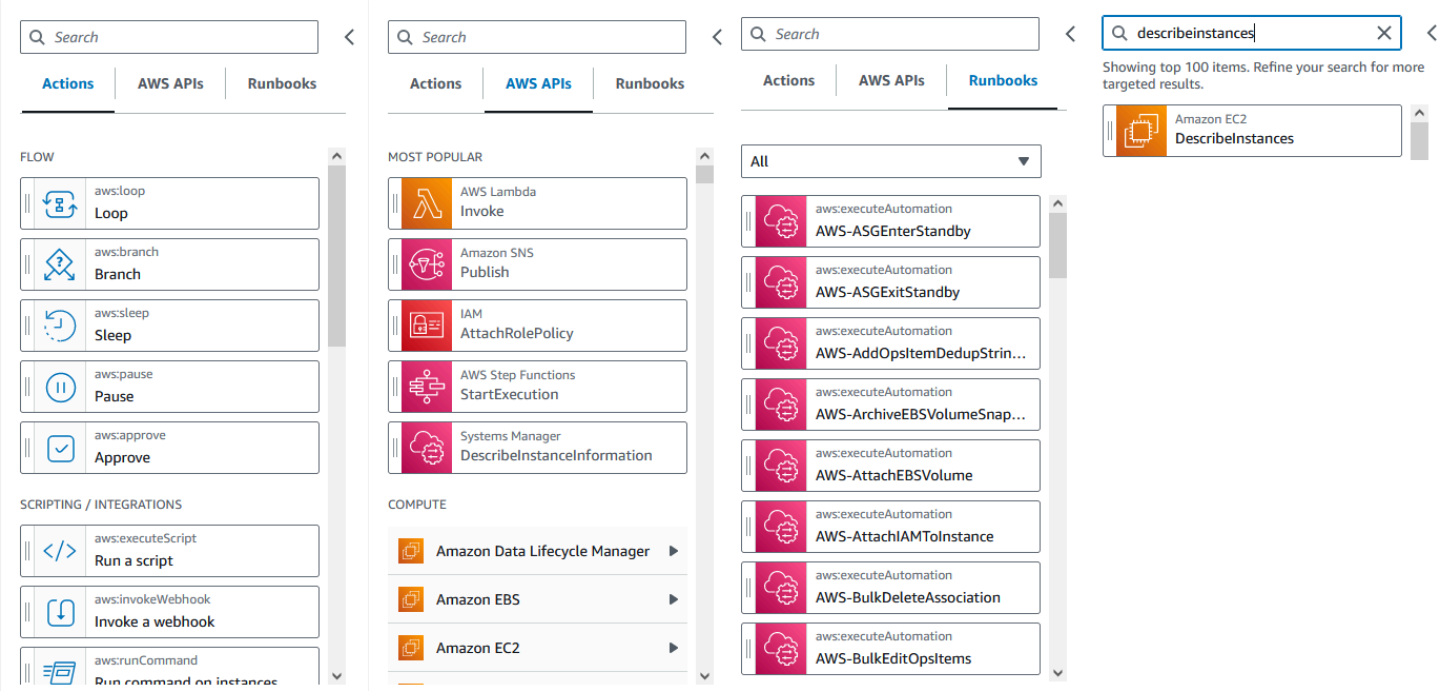
- Le panneau Formulaire vous permet d’afficher et de modifier les propriétés de toute action que vous avez sélectionnée sur le canevas. Sélectionnez le bouton Contenu pour afficher le code YAML ou JSON de votre runbook, l’action actuellement sélectionnée étant surlignée.

Les liens Informations ouvrent un panneau contenant des informations contextuelles lorsque vous avez besoin d’aide. Ces panneaux incluent également des liens vers des sujets connexes dans la documentation de Systems Manager Automation.

Navigateur d’actions

Dans le navigateur Actions, vous pouvez sélectionner des actions à glisser-déposer dans votre graphique de flux de travail. Vous pouvez rechercher toutes les actions à l’aide du champ de recherche situé en haut du navigateur Actions. Le navigateur Actions contient les onglets suivants :

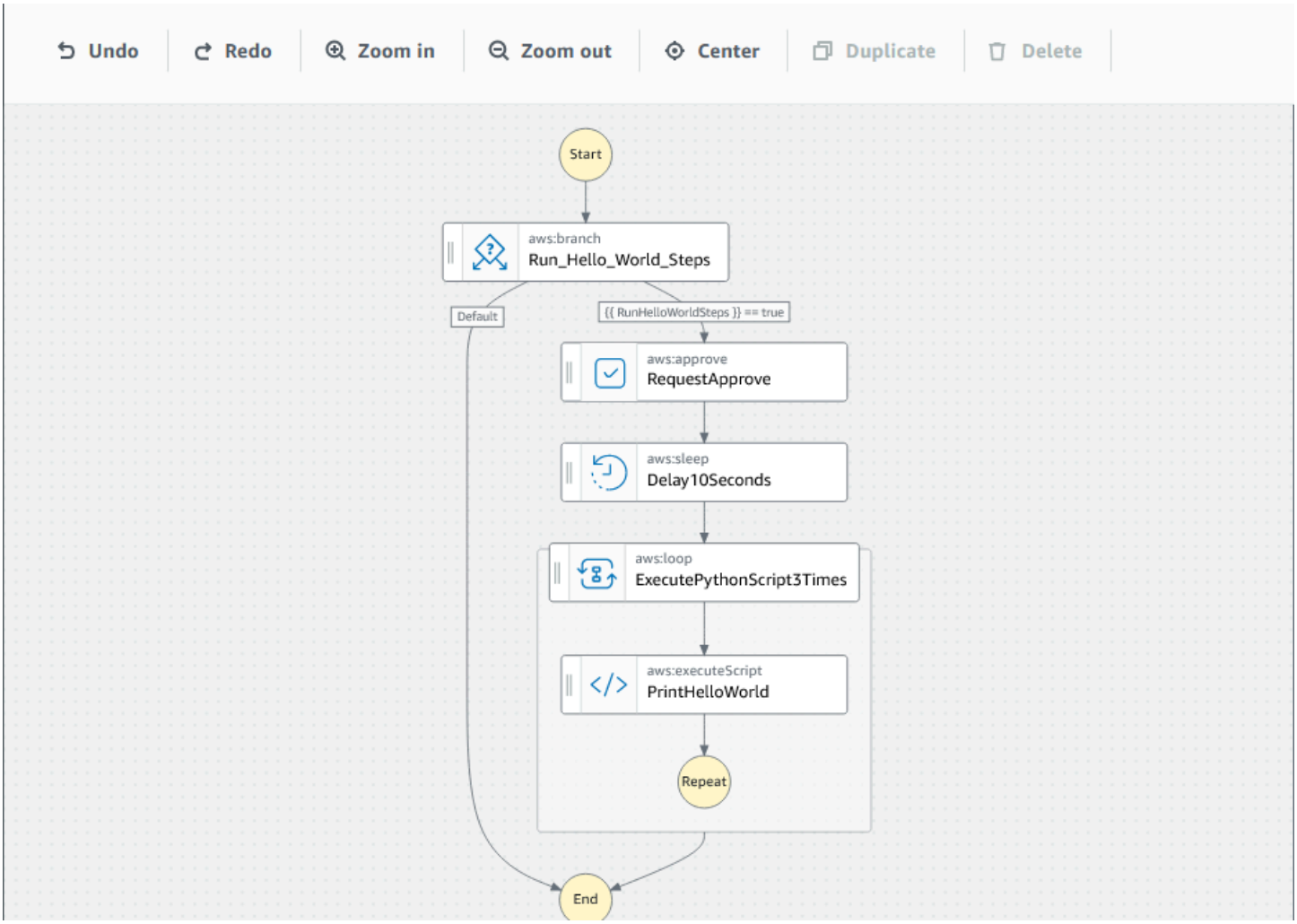
- L’onglet Actions fournit une liste d’actions d’automatisation que vous pouvez glisser-déposer dans le graphique du flux de travail de votre runbook dans le canevas.
- L'onglet AWS API fournit une liste d' AWS API que vous pouvez glisser-déposer dans le graphique du flux de travail de votre runbook dans le canevas.
- L'onglet Runbooks fournit plusieurs ready-to-use runbooks réutilisables sous forme de blocs de construction que vous pouvez utiliser dans divers cas d'utilisation. Par exemple, vous pouvez utiliser des runbook pour effectuer des tâches de correction courantes sur les instances Amazon EC2 de votre flux de travail sans avoir à recréer les mêmes actions.



Canvas

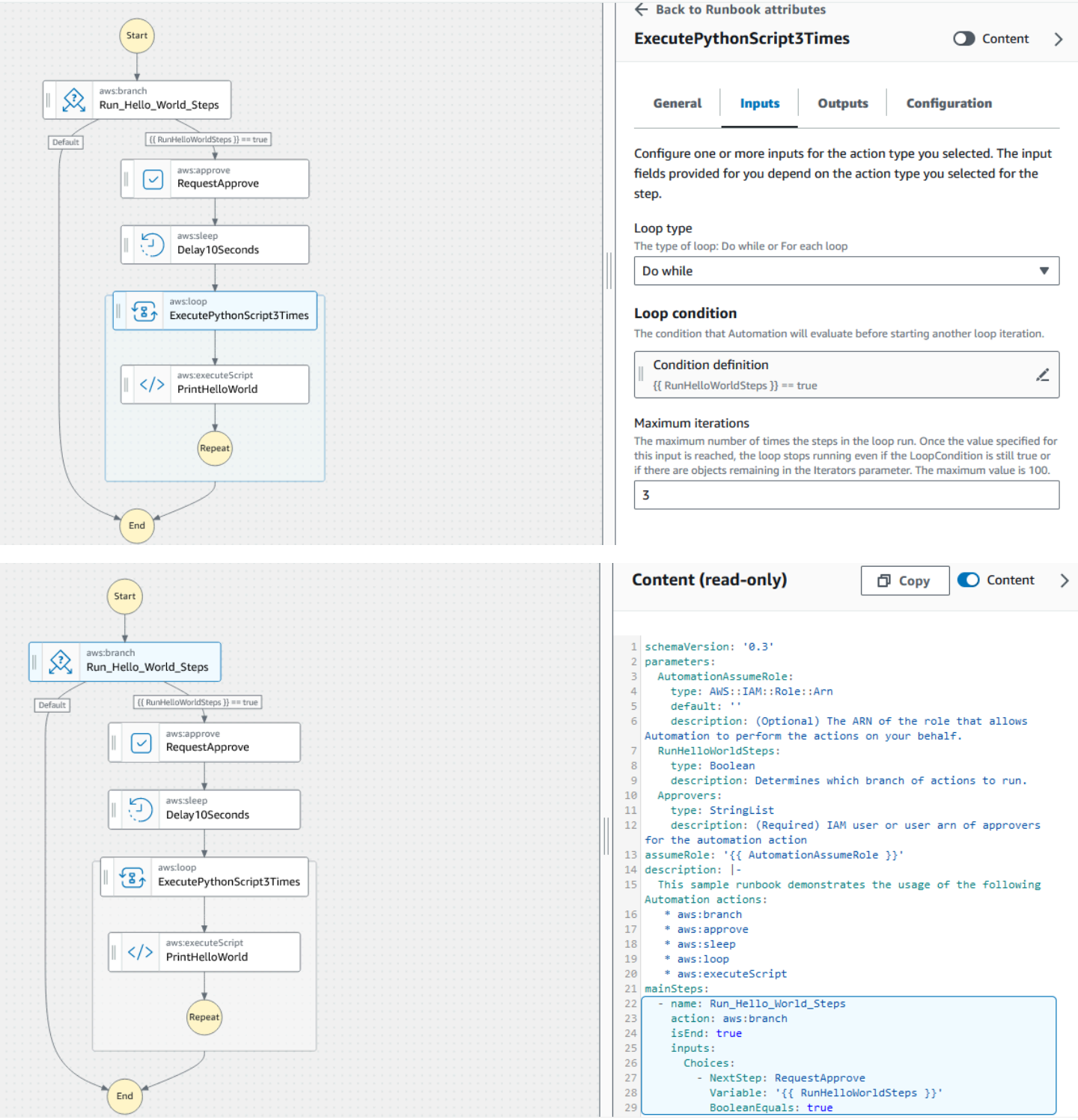
Après avoir choisi une action à ajouter à votre automatisation, faites-la glisser vers le canevas et déposez-la dans votre graphe de flux de travail. Vous pouvez également glisser-déposer des actions pour les déplacer à différents endroits du flux de travail de votre runbook. Si votre flux de travail est complexe, vous pouvez rencontrer des difficultés pour l'afficher dans son intégralité dans le panneau canevas. Utilisez les commandes situées en haut du canevas pour zoomer ou dézoomer. Pour afficher les différentes parties d'un flux de travail, vous pouvez faire glisser le graphique du flux de travail dans le canevas.

Faites glisser une action depuis le navigateur Actions et déposez-la dans le graphique du flux de travail de votre runbook. Une ligne indique où il sera placé dans votre flux de travail. Pour modifier l'ordre d'une action, vous pouvez la faire glisser vers un autre endroit de votre flux de travail. La nouvelle action a été ajoutée à votre flux de travail et son code est généré automatiquement.



Formulaire

Après avoir ajouté une action à votre flux de travail runbook, vous pouvez la configurer en fonction de votre cas d'utilisation. Sélectionnez l'action que vous souhaitez configurer et vous verrez ses paramètres et options dans le panneau Formulaire. Vous pouvez également voir le code YAML ou JSON en choisissant le bouton Contenu. Le code associé à l'action que vous avez sélectionnée est surligné.



Raccourcis clavier

L'expérience de conception visuelle prend en charge les raccourcis clavier indiqués dans le tableau suivant.

Raccourci clavier	Action
Ctrl + Z	Annule la dernière opération.
Ctrl + Maj + Z	Rétablit la dernière opération.
Alt + C	Centre le flux de travail dans le canevas.
Retour arrière	Supprime les états sélectionnés.
Supprimer	Supprime les états sélectionnés.

Raccourci clavier

Duplique l'état sélectionné.

Utilisation de l'expérience de conception visuelle

Apprenez à créer, modifier et exécuter des flux de travail runbook à l'aide de l'expérience de conception visuelle. Une fois que votre flux de travail est prêt, vous pouvez l'enregistrer ou l'exporter. Vous pouvez également utiliser l'expérience de conception visuelle pour un prototypage rapide.

Création d'un flux de travail runbook

1. Connectez-vous à la console Systems Manager Automation.
2. Sélectionnez Créer un runbook.
3. Dans le champ Nom, saisissez le nom de votre runbook, *MyNewRunbook*, par exemple.
4. À côté du bouton Design et Code, sélectionnez l'icône en forme de crayon et saisissez le nom de votre runbook.

Vous pouvez désormais concevoir un flux de travail pour votre nouveau runbook.

Conception d'un runbook

Pour concevoir un flux de travail runbook à l'aide de l'expérience de conception visuelle, vous devez faire glisser une action d'automatisation du navigateur Actions vers le canevas, en la plaçant là où vous le souhaitez dans le flux de travail de votre runbook. Vous pouvez également réorganiser les actions de votre flux de travail en les faisant glisser vers un autre emplacement. Lorsque vous faites glisser une action sur le canevas, une ligne apparaît à l'endroit où vous pouvez déposer l'action dans votre flux de travail. Une fois qu'une action est déposée sur le canevas, son code est généré automatiquement et ajouté au contenu de votre runbook.

Si vous connaissez le nom de l'action que vous souhaitez ajouter, utilisez le champ de recherche en haut du navigateur Actions pour trouver l'action.

Après avoir déposé une action sur le canevas, configurez-la à l'aide du panneau Formulaire sur la droite. Ce panneau contient les onglets Général, Entrées, Sorties et Configuration pour chaque action d'automatisation ou action d'API que vous placez sur le canevas. Par exemple, l'onglet Général comprend les sections suivantes :

- Le Nom de l'étape identifie l'étape. Spécifiez une valeur unique pour le nom de l'étape.
- La Description vous permet de décrire l'action en cours dans le flux de travail de votre runbook.

L'onglet Entrées contient des champs qui varient en fonction de l'action. Par exemple, l'action d'automatisation `aws:executeScript` comprend les sections suivantes :

- L'exécution est le langage à utiliser pour exécuter le script fourni.
- Le gestionnaire est le nom de votre fonction. Vous devez vous assurer que la fonction définie dans le gestionnaire possède deux paramètres : `events` et `context`. L'exécution de PowerShell ne prend pas en charge ce paramètre.
- Script est un script incorporé que vous souhaitez exécuter pendant le flux de travail.
- (Facultatif) La Pièce jointe est destinée aux scripts autonomes ou aux fichiers .zip qui peuvent être invoqués par l'action. Ce paramètre est obligatoire pour les runbook JSON.

L'onglet Sorties vous permet de spécifier les valeurs que vous souhaitez obtenir à partir d'une action. Vous pouvez référencer des valeurs de sortie dans des actions ultérieures de votre flux de travail ou générer des sorties à partir d'actions à des fins de journalisation. Toutes les actions n'auront pas d'onglet Sorties car toutes les actions ne prennent pas en charge les sorties. Par exemple, l'action `aws:pause` ne prend pas en charge les sorties. Pour les actions qui prennent en charge les sorties, l'onglet Sorties comprend les sections suivantes :

- Le Nom est le nom à utiliser pour la valeur de sortie. Vous pouvez référencer les résultats dans des actions ultérieures de votre flux de travail.
- Le Sélecteur est une chaîne d'expression JSONPath commençant par `"$."` qui est utilisée pour sélectionner un ou plusieurs composants au sein d'un élément JSON.
- Le Type est le type de données pour la valeur de sortie. Par exemple, un type de donnée `String` ou `Integer`.

L'onglet Configuration contient des propriétés et des options utilisables par toutes les actions d'automatisation. L'action se compose des sections suivantes :

- La propriété Maximal de tentatives indique le nombre de tentatives répétées d'une action en cas d'échec.
- La propriété Délai en secondes spécifie la valeur du délai d'expiration d'une action.
- La propriété Est critique détermine si l'échec de l'action arrête l'ensemble de l'automatisation.
- La propriété Etape suivante détermine l'action à laquelle l'automatisation passe ensuite dans le runbook.
- La propriété En cas d'échec détermine à quelle action l'automatisation passe ensuite dans le runbook en cas d'échec de l'action.
- La propriété En cas d'annulation détermine à quelle action l'automatisation passe ensuite dans le runbook si l'action est annulée par un utilisateur.

Pour supprimer une action, vous pouvez utiliser l'espace arrière, la barre d'outils au-dessus du canevas ou cliquer avec le bouton droit de la souris et choisir Supprimer l'action.

Au fur et à mesure que votre flux de travail se développe, il se peut qu'il ne rentre pas dans le canevas. Pour adapter le flux de travail au canevas, essayez l'une des options suivantes :

- Utilisez les commandes situées sur les panneaux latéraux pour redimensionner ou fermer les panneaux.
- Utilisez la barre d'outils située en haut du canevas pour zoomer ou dézoomer le graphique du flux de travail.

Mise à jour de votre runbook

Vous pouvez mettre à jour un flux de travail de runbook existant en créant une nouvelle version de votre runbook. Les mises à jour de vos runbook peuvent être effectuées à l'aide de l'expérience de conception visuelle ou en modifiant directement le code. Pour mettre à jour un runbook existant, procédez comme suit :

1. Connectez-vous à la console Systems Manager Automation.
2. Sélectionnez le runbook que vous souhaitez mettre à jour.
3. Choisissez Create new version (Créer une version).
4. L'expérience de conception visuelle comporte deux volets : un volet de code et un volet de flux de travail visuel. Sélectionnez Design dans le volet du flux de travail visuel pour modifier votre flux de travail avec l'expérience de conception visuelle. Lorsque vous avez terminé, sélectionnez Créer une nouvelle version pour enregistrer les modifications et quitter.

5. (Facultatif) Utilisez le volet de code pour modifier le contenu du runbook en YAML ou JSON.

Exportation de votre runbook

Pour exporter le code YAML ou JSON du flux de travail de votre runbook, ainsi qu'un graphique de votre flux de travail, procédez comme suit :

1. Choisissez votre runbook dans la console Documents.
2. Choisissez Create new version (Créer une version).
3. Dans le menu déroulant Actions, sélectionnez si vous souhaitez exporter le diagramme ou le runbook, et sélectionnez le format que vous préférez.

Configurer les entrées et les sorties pour vos actions

Chaque action d'automatisation répond en fonction des entrées qu'elle reçoit. Dans la plupart des cas, vous transmettez ensuite le résultat aux actions suivantes. Dans l'expérience de conception visuelle, vous pouvez configurer les données d'entrée et de sortie d'une action dans les onglets Entrées et Sorties du panneau Formulaire.

Pour obtenir des informations détaillées sur la définition et l'utilisation de la sortie pour les actions d'automatisation, consultez Utilisation des sorties d'action comme entrées.

Fournir des données d'entrée pour une action

Chaque action d'automatisation comporte une ou plusieurs entrées pour lesquelles vous devez fournir une valeur. La valeur que vous fournissez pour la saisie d'une action est déterminée par le type de données et le format acceptés par l'action. Par exemple, les actions `aws:sleep` nécessitent une valeur de chaîne au format ISO 8601 pour l'entrée `Duration`.

Généralement, vous utilisez des actions dans le flux de travail de votre runbook qui renvoient des résultats que vous souhaitez utiliser dans les actions suivantes. Il est important de vous assurer que les valeurs d'entrée sont correctes afin d'éviter des erreurs dans le flux de travail de votre runbook. Les valeurs d'entrée sont également importantes car elles déterminent si l'action renvoie le résultat attendu. Par exemple, lorsque vous utilisez l'action `aws:executeAwsApi`, vous devez vous assurer que vous fournissez la bonne valeur pour l'opération d'API.

Définir les données de sortie pour une action

Certaines actions d'automatisation renvoient un résultat après avoir effectué les opérations définies. Les actions qui renvoient une sortie ont des sorties prédéfinies ou vous permettent de définir vous-même les sorties. Par exemple, l'action `aws:createImage` possède des sorties prédéfinies qui renvoient `ImageId` et `ImageState`. En comparaison, avec l'action `aws:executeAwsApi`, vous pouvez définir les sorties que vous souhaitez obtenir de l'opération d'API spécifiée. Par conséquent, vous pouvez renvoyer une ou plusieurs valeurs issues d'une seule opération d'API à utiliser dans les actions suivantes.

Pour définir vos propres sorties pour une action d'automatisation, vous devez spécifier le nom de la sortie, le type de données et la valeur de sortie. Pour continuer à utiliser l'action `aws:executeAwsApi` comme exemple, supposons que vous appelez l'opération d'API de`DescribeInstances` depuis Amazon EC2. Dans cet exemple, vous souhaitez renvoyer, ou générer, le `State` d'une instance Amazon EC2 et diviser le flux de travail de votre runbook en fonction de la sortie. Vous choisissez de nommer la sortie **`InstanceState`** et d'utiliser le type de données **`String`**.

Le processus de définition de la valeur réelle de la sortie varie en fonction de l'action. Par exemple, si vous utilisez l'action `aws:executeScript`, vous devez utiliser des déclarations `return` dans vos fonctions pour fournir des données à vos sorties. Avec d'autres actions telles que `aws:executeAwsApi`, `aws:waitForAwsResourceProperty` et `aws:assertAwsResourceProperty`, un `Selector` est requis. Le `Selector`, ou `PropertySelector` comme l'appellent certaines actions, est une chaîne JSONPath utilisée pour traiter la réponse JSON d'une opération d'API. Il est important de comprendre comment l'objet de réponse JSON issu d'une opération d'API est structuré afin de pouvoir sélectionner la valeur correcte pour votre sortie. À l'aide de l'opération d'API `DescribeInstances` mentionnée précédemment, consultez l'exemple de réponse JSON suivant :

```
{
  "reservationSet": {
    "item": {
      "reservationId": "r-1234567890abcdef0",
      "ownerId": 123456789012,
      "groupSet": "",
      "instancesSet": {
        "item": {
          "instanceId": "i-1234567890abcdef0",
          "imageId": "ami-bff32ccc",
          "instanceState": {
```

```
            "code": 16,
            "name": "running"
          },
          "privateDnsName": "ip-192-168-1-88.eu-west-1.compute.internal",
          "dnsName": "ec2-54-194-252-215.eu-west-1.compute.amazonaws.com",
          "reason": "",
          "keyName": "my_keypair",
          "amiLaunchIndex": 0,
          "productCodes": "",
          "instanceType": "t2.micro",
          "launchTime": "2018-05-08T16:46:19.000Z",
          "placement": {
            "availabilityZone": "eu-west-1c",
            "groupName": "",
            "tenancy": "default"
          },
          "monitoring": {
            "state": "disabled"
          },
          "subnetId": "subnet-56f5f000",
          "vpcId": "vpc-11112222",
          "privateIpAddress": "192.168.1.88",
          "ipAddress": "54.194.252.215",
          "sourceDestCheck": true,
          "groupSet": {
            "item": {
              "groupId": "sg-e4076000",
              "groupName": "SecurityGroup1"
            }
          },
          "architecture": "x86_64",
          "rootDeviceType": "ebs",
          "rootDeviceName": "/dev/xvda",
          "blockDeviceMapping": {
            "item": {
              "deviceName": "/dev/xvda",
              "ebs": {
                "volumeId": "vol-1234567890abcdef0",
                "status": "attached",
                "attachTime": "2015-12-22T10:44:09.000Z",
                "deleteOnTermination": true
              }
            }
          },
```

```
          "virtualizationType": "hvm",
          "clientToken": "xMcwG14507example",
          "tagSet": {
            "item": {
              "key": "Name",
              "value": "Server_1"
            }
          },
          "hypervisor": "xen",
          "networkInterfaceSet": {
            "item": {
              "networkInterfaceId": "eni-551ba000",
              "subnetId": "subnet-56f5f000",
              "vpcId": "vpc-11112222",
              "description": "Primary network interface",
              "ownerId": 123456789012,
              "status": "in-use",
              "macAddress": "02:dd:2c:5e:01:69",
              "privateIpAddress": "192.168.1.88",
              "privateDnsName": "ip-192-168-1-88.eu-west-1.compute.internal",
              "sourceDestCheck": true,
              "groupSet": {
                "item": {
                  "groupId": "sg-e4076000",
                  "groupName": "SecurityGroup1"
                }
              },
              "attachment": {
                "attachmentId": "eni-attach-39697adc",
                "deviceIndex": 0,
                "status": "attached",
                "attachTime": "2018-05-08T16:46:19.000Z",
                "deleteOnTermination": true
              },
              "association": {
                "publicIp": "54.194.252.215",
                "publicDnsName": "ec2-54-194-252-215.eu-west-1.compute.amazonaws.com",
                "ipOwnerId": "amazon"
              },
              "privateIpAddressesSet": {
                "item": {
                  "privateIpAddress": "192.168.1.88",
                  "privateDnsName": "ip-192-168-1-88.eu-west-1.compute.internal",
                  "primary": true,
```

```
                  "association": {
                    "publicIp": "54.194.252.215",
                    "publicDnsName": "ec2-54-194-252-215.eu-
west-1.compute.amazonaws.com",
                    "ipOwnerId": "amazon"
                  }
                }
              },
              "ipv6AddressesSet": {
                "item": {
                  "ipv6Address": "2001:db8:1234:1a2b::123"
                }
              }
            }
          },
          "iamInstanceProfile": {
            "arn": "arn:aws:iam::123456789012:instance-profile/AdminRole",
            "id": "ABCAJEDNCAA64SSD123AB"
          },
          "ebsOptimized": false,
          "cpuOptions": {
            "coreCount": 1,
            "threadsPerCore": 1
          }
        }
      }
    }
  }
}
```

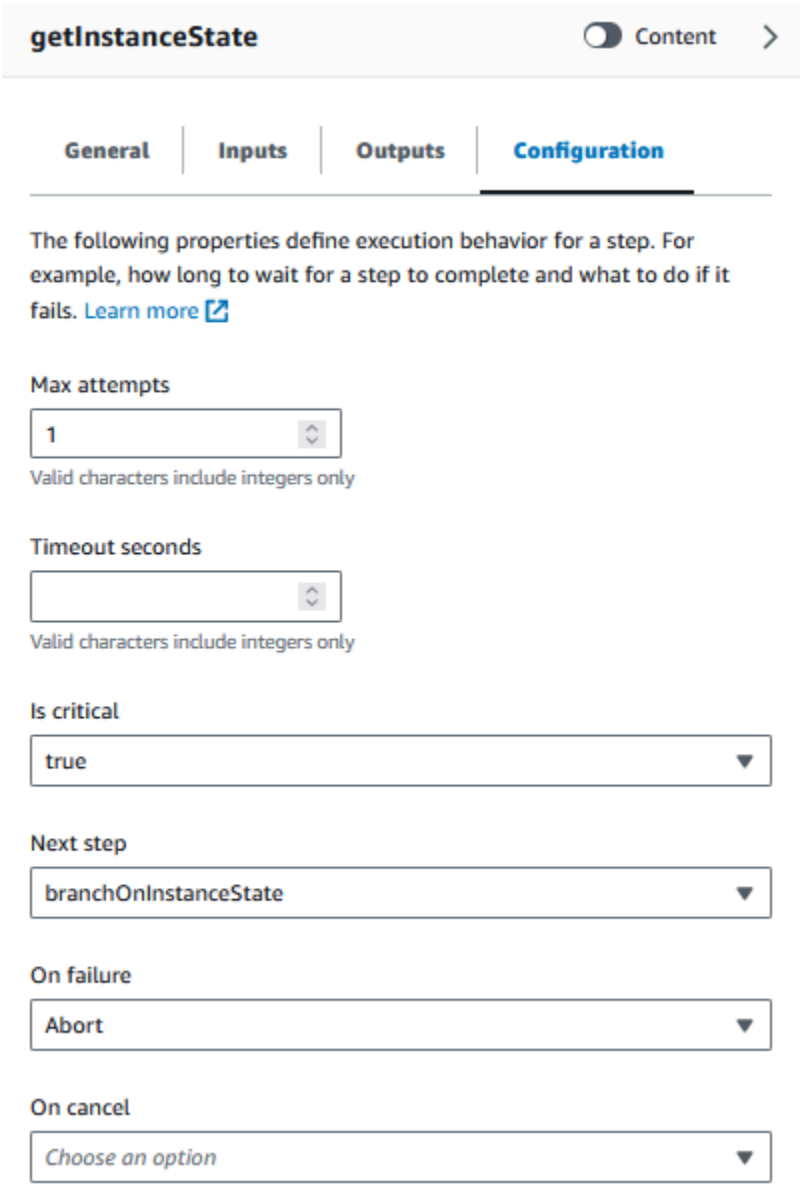
Dans l'objet de réponse JSON, l'instance `State` est imbriquée dans un objet `Instances`, qui est imbriqué dans l'objet `Reservations`. Pour renvoyer la valeur de l'instance `State`, utilisez la chaîne suivante pour le `Selector` de sorte que la valeur puisse être utilisée dans notre sortie : **`$.Reservations[0].Instances[0].State.Name`**.

Pour référencer une valeur de sortie dans les actions suivantes du flux de travail de votre runbook, le format suivant est utilisé : `{{ StepName.NameOfOutput }}`. Par exemple, **`{{ GetInstanceState.InstanceState }}`**. Dans l'expérience de conception visuelle, vous pouvez choisir les valeurs de sortie à utiliser dans les actions suivantes à l'aide de la liste déroulante des entrées. Lorsque vous utilisez des sorties dans des actions ultérieures, le type de données de la sortie doit correspondre au type de données de l'entrée. Dans cet exemple, la sortie

`InstanceState` est un `String`. Par conséquent, pour utiliser la valeur dans l'entrée d'une action ultérieure, celle-ci doit accepter un `String`.

Gestion des erreurs grâce à l'expérience de conception visuelle

Par défaut, lorsqu'une action signale une erreur, Automation arrête complètement le flux de travail du runbook. Cela est dû au fait que la valeur par défaut de la propriété `onFailure` pour toutes les actions est `Abort`. Vous pouvez configurer la façon dont l'Automatisation gère les erreurs dans le flux de travail de votre runbook. Même si vous avez configuré la gestion des erreurs, certaines erreurs peuvent tout de même entraîner l'échec d'une automatisation. Pour de plus amples informations, veuillez consulter Résolution des problèmes liés à Systems Manager Automation. Dans l'expérience de conception visuelle, vous configurez la gestion des erreurs dans le panneau Configuration.



Réessayer l'action en cas d'erreur

Pour réessayer une action en cas d'erreur, spécifiez une valeur pour la propriété Tentatives maximum. La valeur par défaut est 1. Si vous spécifiez une valeur supérieure à 1, l'action n'est pas considérée comme ayant échoué tant que toutes les tentatives n'ont pas échoué.

Délais

Vous pouvez configurer un délai d'attente pour les actions afin de définir le nombre maximum de secondes pendant lesquelles votre action peut s'exécuter avant d'échouer. Pour configurer un délai d'attente, entrez le nombre de secondes pendant lesquelles votre action doit attendre avant qu'elle n'échoue dans la propriété Délai en secondes. Si le délai est expiré et que l'action a une valeur de

`Max attempts` qui est supérieure à 1, alors l'étape n'est pas considérée comme ayant expirée jusqu'à ce que les tentatives soient terminées.

Actions ayant échoué

Par défaut, lorsqu'une action échoue, Automation arrête complètement le flux de travail du runbook. Vous pouvez modifier ce comportement en spécifiant une valeur alternative pour la propriété En cas d'échec des actions de votre runbook. Si vous souhaitez que le flux de travail passe à l'étape suivante du runbook, choisissez Continuer. Si vous souhaitez que le flux de travail passe à une autre étape ultérieure du runbook, choisissez Étape, puis entrez le nom de l'étape.

Actions annulées

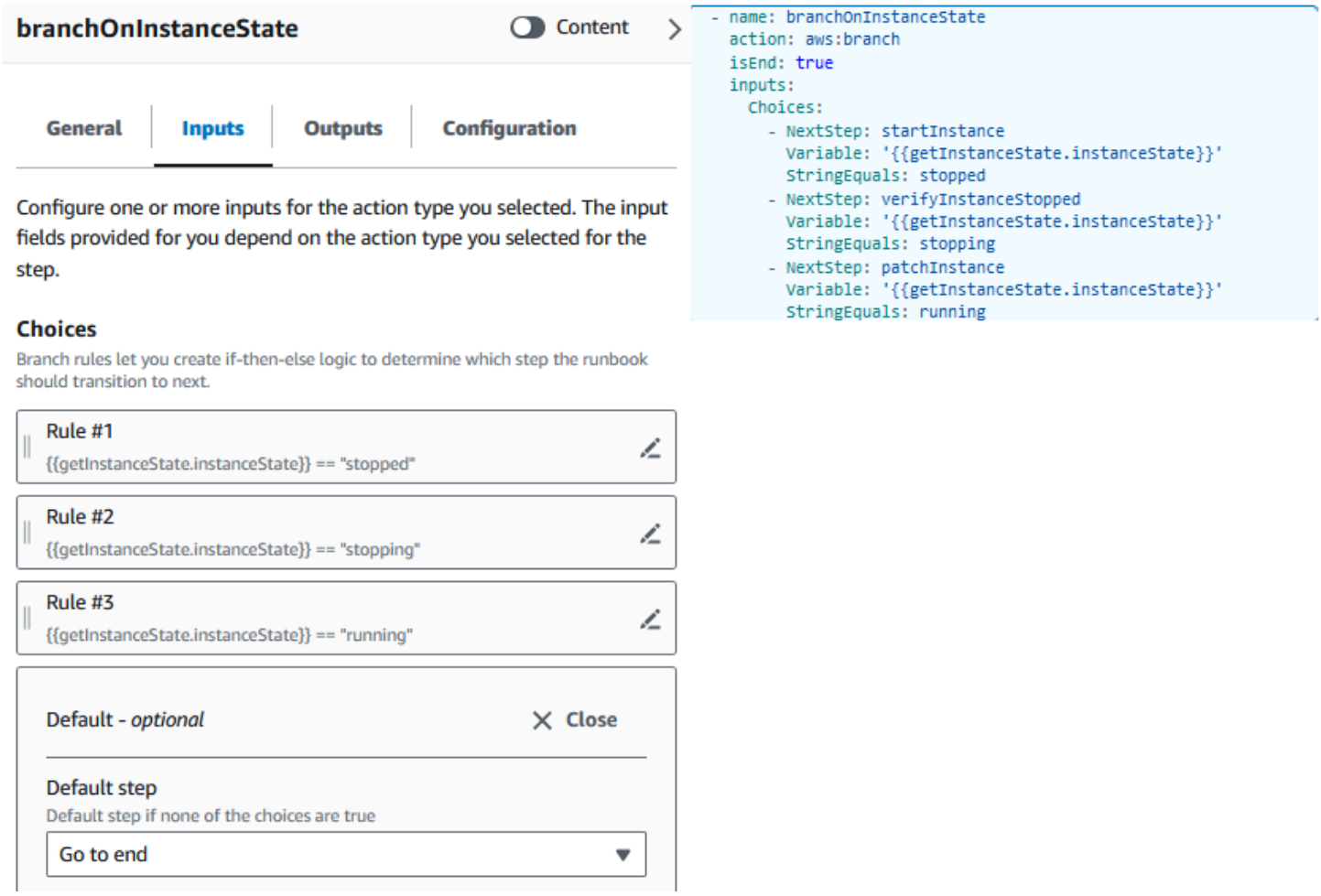
Par défaut, lorsqu'une action est annulée par un utilisateur, Automation arrête complètement le flux de travail du runbook. Vous pouvez modifier ce comportement en spécifiant une valeur alternative pour la propriété Annuler des actions de votre runbook. Si vous souhaitez que le flux de travail passe à une autre étape ultérieure du runbook, choisissez Étape, puis entrez le nom de l'étape.

Actions critiques

Vous pouvez désigner une action comme critique, ce qui signifie qu'elle détermine le statut global des rapports de votre automatisation. Si une étape portant cette désignation échoue, Automation signale l'état final comme `Failed` indépendamment du succès des autres actions. Pour configurer une action comme critique, laissez la valeur par défaut comme Vrai pour la propriété Est critique.

Fin des actions

La propriété Est fin arrête une exécution d'automatisation à la fin de l'action spécifiée. La valeur par défaut pour cette propriété est `false`. Si vous configurez cette propriété pour une action, l'automatisation s'arrête, que l'action réussisse ou échoue. Cette propriété est le plus souvent utilisée avec des actions `aws:branch` destinées à gérer des valeurs d'entrée inattendues ou non définies. L'exemple suivant montre un runbook qui attend un état d'instance tel que `running`, `stopping`, ou `stopped`. Si une instance est dans un état différent, l'automatisation prend fin.



Tutoriel : Création d'un runbook à l'aide de l'expérience de conception visuelle

Dans ce tutoriel, vous apprendrez les bases de l'utilisation de l'expérience de conception visuelle fournie par Systems Manager Automation. Dans le cadre de l'expérience de conception visuelle, vous pouvez créer un runbook qui utilise plusieurs actions. Vous utilisez la fonctionnalité glisser-déposer pour organiser les actions sur le canevas. Vous pouvez également rechercher, sélectionner et configurer ces actions. Vous pouvez ensuite afficher le code YAML généré automatiquement pour le flux de travail de votre runbook, quitter l'expérience de conception visuelle, exécuter le runbook et consulter les détails d'exécution.

Ce tutoriel explique également comment mettre à jour le runbook et afficher la nouvelle version. À la fin du tutoriel, vous effectuez une étape de nettoyage et supprimez votre runbook.

Après avoir terminé ce tutoriel, vous saurez comment utiliser l'expérience de conception visuelle pour créer un runbook. Vous saurez également comment mettre à jour, exécuter et supprimer votre runbook.

> **Note**
>
> Avant de commencer ce tutoriel, assurez-vous d'avoir terminé Configuration d'Automation.

Rubriques

Étape 1 : Navigation vers l'expérience de conception visuelle

1. Connectez-vous à la console Systems Manager Automation.
2. Sélectionnez Créer un runbook d'Automatisation.

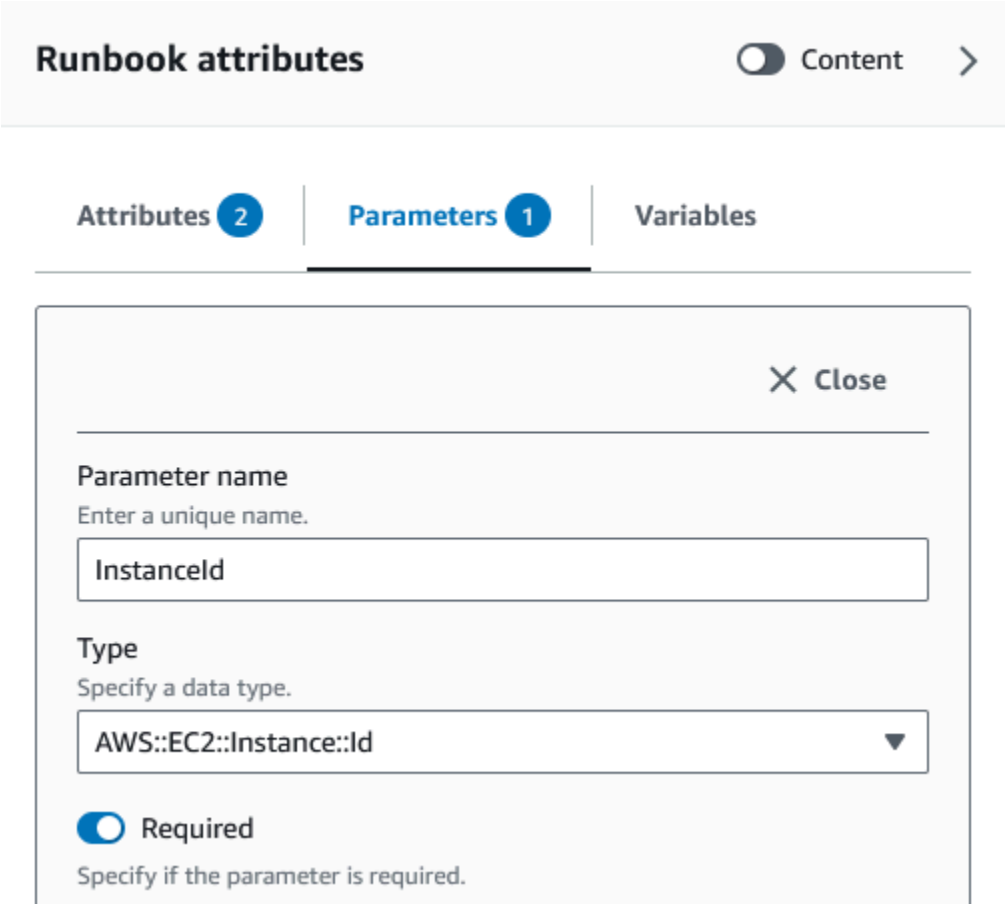
Étape 2 : Création d'un flux de travail

Dans l'expérience de conception visuelle, un flux de travail est une représentation graphique de votre runbook sur le canevas. Vous pouvez utiliser l'expérience de conception visuelle pour définir, configurer et examiner les actions individuelles de votre runbook.

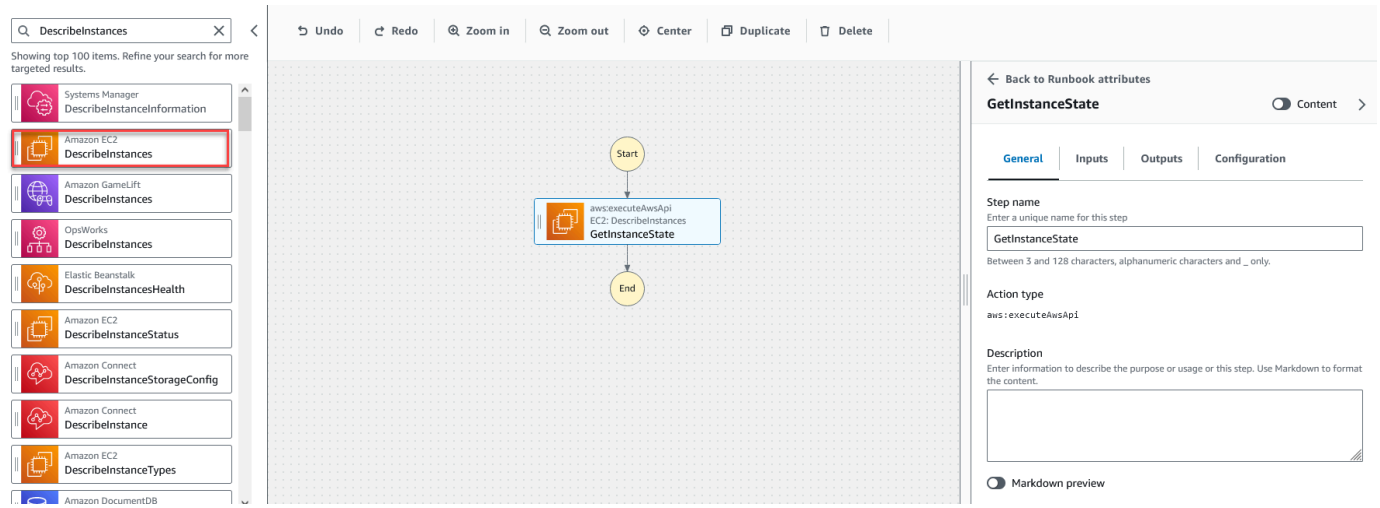
Pour créer un flux de travail

1. À côté du bouton Design et Code, sélectionnez l'icône en forme de crayon et saisissez le nom de votre runbook. Dans le cadre de ce didacticiel, entrez **`VisualDesignExperienceTutorial`**.

2. Dans la section Attributs du document du panneau Formulaire, développez le menu déroulant Paramètres d'entrée, puis sélectionnez Ajouter un paramètre.
   a. Dans le champ Nom du paramètre, saisissez **`InstanceId`**.
   b. Dans le menu déroulant Type, sélectionnez AWS::EC2::Instance.
   c. Sélectionnez le bouton Obligatoire.



3. Dans le navigateur des API AWS , saisissez **`DescribeInstances`** dans la barre de recherche.
4. Faites glisser une DescribeInstances action Amazon EC2 vers le canevas vide.
5. Pour Nom étape, saisissez une valeur. Pour ce tutoriel, vous pouvez utiliser le nom **`GetInstanceState`**.



a. Développez le menu déroulant Entrées supplémentaires, puis dans le champ Nom de l'entrée, saisissez **`InstanceIds`**.

b. Sélectionnez l'onglet Entrées.

c. Dans le champ Valeur d'entrée, sélectionnez l'entrée du document **`InstanceId`**. Cela fait référence à la valeur du paramètre d'entrée que vous avez créé au début de la procédure. Étant donné que l'InstanceIdsentrée de l'`DescribeInstances`action accepte `StringList` des valeurs, vous devez placer l'InstanceIdentrée entre crochets. Le YAML pour la valeur d'entrée doit correspondre à ce qui suit : **`['{{ InstanceId }}']`**

d. Dans l'onglet Sorties, sélectionnez Ajouter une sortie et saisissez **`InstanceState`** dans le champ Nom.

e. Dans le champ Sélecteur, saisissez **`$.Reservations[0].Instances[0].State.Name`**.

f. Dans le menu déroulant Type, sélectionnez String.

6. Faites glisser une action Branche depuis le navigateur Actions, puis déposez-la en dessous de l'étape **`GetInstanceState`**.

7. Pour Nom étape, saisissez une valeur. Pour ce tutoriel, utilisez le nom **`BranchOnInstanceState`**.

   Pour définir la logique de branchement, procédez comme suit :

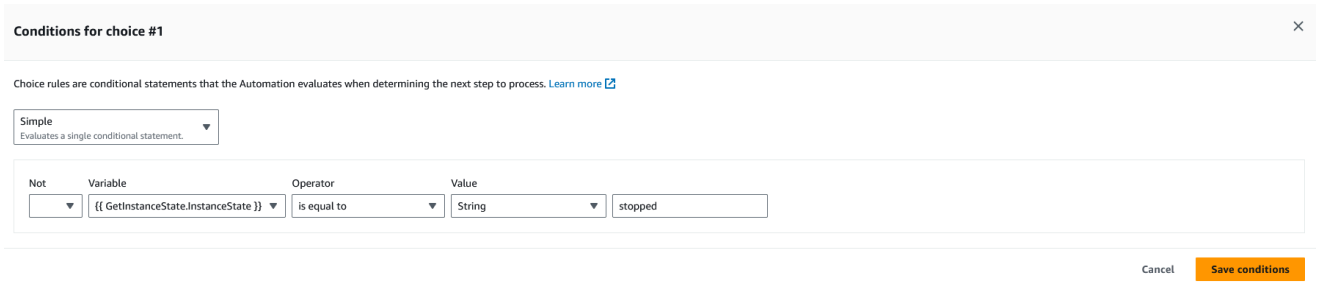
   a. Choisissez l'état **`Branch`** sur le canevas. Ensuite, sous Entrées et Choix, sélectionnez l'icône en forme de crayon pour modifier la Règle n°1.

   b. Sélectionnez Ajouter des conditions.

   c. Dans la boîte de dialogue Conditions pour la règle n°1, sélectionnez le résultat de l'étape **`GetInstanceState.InstanceState`** dans le menu déroulant Variable.

   d. Pour Opérateur, sélectionnez est égal à.

   e. Pour Valeur, sélectionnez String dans la liste déroulante. Saisissez **`stopped`**.



   f. Sélectionnez Enregistrer les conditions.

g. Sélectionnez Ajouter une nouvelle règle de choix.

h. Sélectionnez Ajouter des conditions pour la Règle n°2.

i. Dans la boîte de dialogue Conditions pour la règle n°2, sélectionnez le résultat de l'étape **`GetInstanceState.InstanceState`** dans le menu déroulant Variable.

j. Pour Opérateur, sélectionnez est égal à.

k. Pour Valeur, sélectionnez String dans la liste déroulante. Saisissez **`stopping`**.

l. Sélectionnez Enregistrer les conditions.

m. Sélectionnez Ajouter une nouvelle règle de choix.

n. Pour la Règle n°3, sélectionnez Ajouter des conditions.

o. Dans la boîte de dialogue Conditions pour la règle n°3, sélectionnez le résultat de l'étape **`GetInstanceState.InstanceState`** dans le menu déroulant Variable.

p. Pour Opérateur, sélectionnez est égal à.

q. Pour Valeur, sélectionnez String dans la liste déroulante. Saisissez **`running`**.

r. Sélectionnez Enregistrer les conditions.

s. Dans la Règle par défaut, sélectionnez Aller à la fin pour l'Etape par défaut.

8. Faites glisser une action Modifier l'état d'une instance vers la zone vide Faire glisser l'action ici sous le {{ GetInstanceState. InstanceState }} == Condition « arrêtée ».

   a. Pour le Nom de l'étape, saisissez **`StartInstance`**.

   b. Dans l'onglet Entrées, sous ID d'instance, choisissez la valeur d'entrée du InstanceIddocument dans le menu déroulant.

   c. Pour l'Etat souhaité, spécifiez **`running`**.

9. Faites glisser une action Attendre la AWS ressource vers la zone vide Faire glisser l'action ici sous le {{ GetInstanceState. InstanceState }} == condition « d'arrêt ».

10. Pour Nom étape, saisissez une valeur. Pour ce tutoriel, utilisez le nom **`WaitForInstanceStop`**.

    a. Pour le champ Service, sélectionnez Amazon EC2.

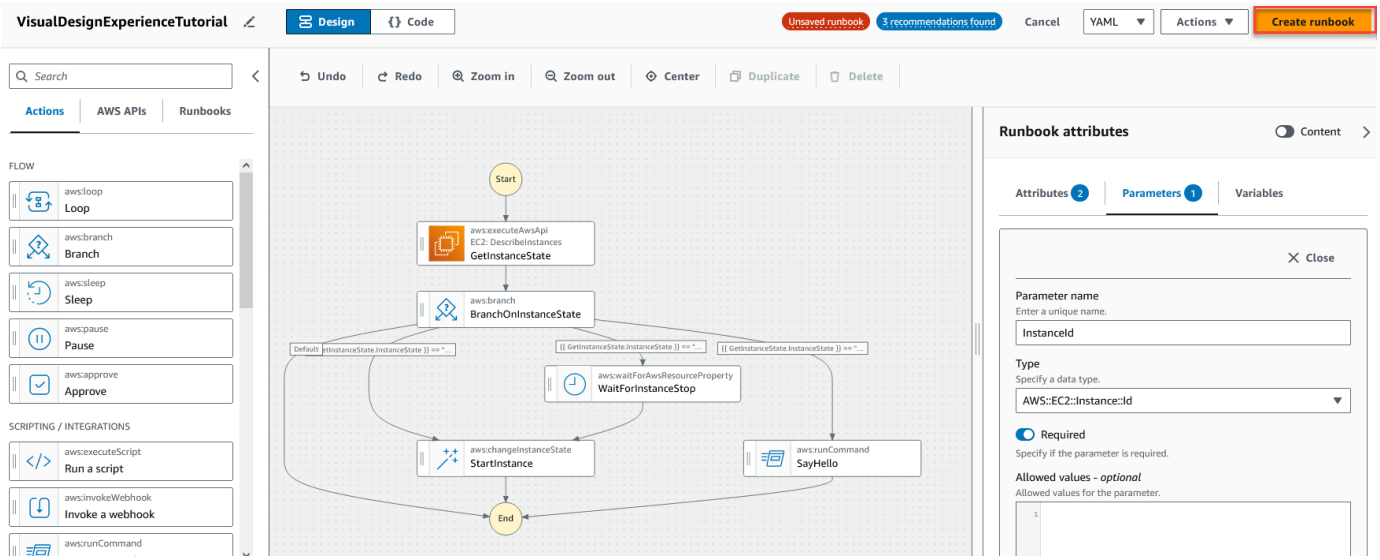
    b. Pour le champ API, sélectionnez DescribeInstances.

    c. Pour le champ Sélecteur de propriétés, saisissez **`$.Reservations[0].Instances[0].State.Name`**.

    d. Pour le paramètre Valeurs souhaitées, saisissez **`["stopped"]`**.

    e. Dans l'onglet Configuration de l'WaitForInstanceStopaction, choisissez dans le StartInstancemenu déroulant Étape suivante.

11. Faites glisser une action Exécuter la commande sur les instances vers la zone vide Drag action here située sous le {{ GetInstanceState. InstanceState }} == condition « en cours d'exécution ».
12. Pour le Nom de l'étape, saisissez **`SayHello`**.
    a. Dans l'onglet Entrées, saisissez **`AWS-RunShellScript`** pour le paramètre Nom du document.
    b. Pour InstanceIds, choisissez la valeur d'entrée du InstanceIddocument dans la liste déroulante.
    c. Développez le menu déroulant Entrées supplémentaires, puis sélectionnez Paramètres dans le menu déroulant Nom de l'entrée.
    d. Saisissez **`{"commands": "echo 'Hello World'"}`** dans le champ Valeur de l'entrée.
13. Passez en revue le runbook terminé dans le canevas et sélectionnez Créer un runbook pour enregistrer le runbook du tutoriel.



Étape 3 : Vérification du code généré automatiquement

Lorsque vous glissez et déposez des actions depuis le navigateur Actions vers le canevas, l'expérience de conception visuelle compose automatiquement le contenu YAML ou JSON de votre runbook en temps réel. Vous pouvez consulter et modifier ce code. Pour afficher le code généré automatiquement, sélectionnez Code pour le commutateur Conception et Code.

Étape 4 : Exécution de votre nouveau runbook

Après avoir créé votre runbook, vous pouvez exécuter l'automatisation.

Pour exécuter votre nouveau runbook d'automatisation

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Automation (Automatisation), puis Execute automation (Exécuter l'automatisation).
3. Dans la liste Automation document (Document Automation), sélectionnez un runbook. Sélectionnez une ou plusieurs options dans le panneau Document categories (Catégories de documents) pour filtrer les documents SSM en fonction de leur but. Pour afficher un runbook vous appartenant, sélectionnez l'onglet Owned by me (M'appartenant). Pour afficher un runbook partagé avec votre compte, sélectionnez l'onglet Shared with me (Partagé avec moi). Pour afficher tous les runbooks, sélectionnez l'onglet All documents (Tous les documents).

   > **Note**
   >
   > Vous pouvez consulter les informations sur un runbook en sélectionnant 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 :

   - Version par défaut lors de l'exécution : sélectionnez cette option si le runbook d'Automatisation est mis à jour régulièrement et qu'une nouvelle version par défaut est attribuée.
   - Dernière version lors de l'exécution : sélectionnez cette option si le runbook d'Automatisation est mis à jour régulièrement et que vous souhaitez exécuter la dernière version mise à jour.
   - 1 (Par défaut) : sélectionnez cette option pour exécuter la première version du document, qui est la version par défaut.

5. Choisissez Suivant.
6. Dans la section Exécuter le runbook d'Automatisation, sélectionnez Exécution simple.
7. Dans la section Input parameters (Paramètres d'entrée), spécifiez les entrées obligatoires. Vous pouvez éventuellement choisir un rôle de service IAM dans la AutomationAssumeRoleliste.
8. (Facultatif) Choisissez une CloudWatch alarme Amazon à appliquer à votre automatisation à des fins de surveillance. Pour associer une CloudWatch alarme à votre automatisation, le principal IAM qui lance l'automatisation doit être autorisé à effectuer l'`iam:createServiceLinkedRole`action. Pour plus d'informations sur les CloudWatch alarmes, consultez la section Utilisation des CloudWatch alarmes Amazon. Si votre alarme

s'active, l'automatisation s'arrête. Si vous utilisez AWS CloudTrail, vous verrez l'appel d'API dans votre journal.

9. Sélectionnez Execute (Exécuter).

Étape 5 : nettoyer

Pour supprimer votre runbook

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.
3. Sélectionnez l'onglet M'appartenant.
4. Localisez le VisualDesignExperienceTutorialrunbook.
5. Sélectionnez le bouton sur la page de la fiche du document, puis sélectionnez Supprimer le document dans le menu déroulant Actions.

Création de runbooks Automation

Chaque runbook d'Automation, une fonctionnalité de AWS Systems Manager, définit une automatisation. Les runbooks Automation définissent les actions effectuées durant une automatisation. Dans le contenu du runbook, vous définissez les paramètres d'entrée, les sorties et les actions que Systems Manager exécute sur vos instances et AWS ressources gérées.

L'automatisation inclut plusieurs runbooks prédéfinis que vous pouvez utiliser afin d'effectuer des tâches courantes, telles que le redémarrage d'une ou plusieurs instances Amazon Elastic Compute Cloud (Amazon EC2) ou la création d'une Amazon Machine Image (AMI). Cependant, vos cas d'utilisation peuvent dépasser les fonctionnalités des runbooks prédéfinis. Dans ce cas, vous pouvez créer vos propres runbooks et les modifier en fonction de vos besoins.

Un runbook se compose d'actions d'automatisation, de paramètres relatifs à ces actions et de paramètres d'entrée que vous spécifiez. Le contenu d'un runbook est écrit en YAML ou en JSON. Si vous ne connaissez ni le YAML ni le JSON, nous vous recommandons d'utiliser le concepteur visuel ou d'en apprendre davantage sur l'un ou l'autre des langages de balisage avant d'essayer de créer votre propre runbook. Pour plus d'informations sur le concepteur visuel, consultezExpérience de conception visuelle pour les runbook d'automatisation.

Les sections suivantes vous aideront à créer votre premier runbook.

Identifier votre cas d'utilisation

La première étape de la création d'un runbook consiste à identifier votre cas d'utilisation. Par exemple, vous avez planifié le runbook `AWS-CreateImage` pour qu'il s'exécute quotidiennement sur toutes vos instances Amazon EC2 de production. À la fin du mois, vous décidez d'avoir plus d'images que nécessaire pour les points de récupération. Ensuite, vous voulez supprimer automatiquement l'AMI la plus ancienne d'une instance Amazon EC2 lorsqu'une nouvelle AMI est créée. Pour cela, vous créez un runbook qui :

1. Exécute l'action `aws:createImage` et spécifie l'ID d'instance dans la description de l'image.
2. Exécute l'action `aws:waitForAwsResourceProperty` pour interroger le statut de l'image jusqu'à ce qu'elle soit `available`.
3. Une fois que l'image a le statut `available`, l'action `aws:executeScript` exécute un script Python personnalisé qui rassemble les ID de toutes les images associées à votre instance Amazon EC2. Pour cela, le script effectue un filtrage à l'aide de l'ID d'instance dans la description de l'image que vous avez spécifiée lors de la création. Ensuite, le script trie la liste des ID d'image en fonction de la `creationDate` de l'image et génère l'ID de l'AMI la plus ancienne.
4. Enfin, l'action `aws:deleteImage` s'exécute pour supprimer l'AMI la plus ancienne à l'aide de l'ID de la sortie de l'étape précédente.

Dans ce scénario, vous utilisiez déjà le runbook `AWS-CreateImage`, mais vous avez constaté que votre cas d'utilisation nécessitait une plus grande flexibilité. C'est une situation courante en raison de l'éventualité d'un chevauchement entre les runbooks et les actions d'automatisation. Par conséquent, vous devrez peut-être ajuster les runbooks ou les actions à utiliser pour traiter votre cas d'utilisation.

Par exemple, les actions `aws:executeScript` et `aws:invokeLambdaFunction` vous permettent d'exécuter des scripts personnalisés dans le cadre de votre automatisation. Entre les deux, `aws:invokeLambdaFunction` peut vous sembler préférable du fait qu'elle prend en charge des langages d'exécution supplémentaires. Pour sa part, `aws:executeScript` vous permet de créer votre contenu de script directement dans les runbooks YAML et de fournir du contenu de script sous forme de pièces jointes pour les runbooks JSON. Vous pouvez même considérer que l'action `aws:executeScript` est plus simple en termes de configuration d' AWS Identity and Access Management (IAM). Comme il utilise les autorisations fournies dans le`AutomationAssumeRole`, il `aws:executeScript` ne nécessite pas de rôle d'exécution de AWS Lambda fonction supplémentaire.

Dans un scénario donné, une action peut offrir plus de flexibilité, ou des fonctionnalités supplémentaires, par rapport à l'autre. Nous vous recommandons donc d'examiner les paramètres d'entrée disponibles pour le runbook ou l'action à utiliser afin de déterminer celui qui correspond le mieux à votre cas d'utilisation et à vos préférences.

Configurer votre environnement de développement.

Après avoir identifié votre cas d'utilisation et les runbooks ou les actions d'automatisation prédéfinis à utiliser dans votre runbook, vous devez configurer votre environnement de développement pour le contenu de votre runbook. Pour développer le contenu de votre runbook, nous vous recommandons d'utiliser la console Documents AWS Toolkit for Visual Studio Code au lieu de la console Systems Manager Documents.

Le Toolkit for VS Code est une extension open source de Visual Studio Code (VS Code), qui offre davantage de fonctionnalités que la console de documents Systems Manager. Les fonctionnalités utiles incluent la validation de schéma pour YAML et JSON, des extraits pour les types d'action d'automatisation et la prise en charge de la saisie automatique de diverses options dans YAML et JSON.

Pour de plus amples informations sur l'installation du Toolkit for VS Code, veuillez consulter Installation du AWS Toolkit for Visual Studio Code. Pour plus d'informations sur l'utilisation de Toolkit for VS Code pour développer des runbooks, veuillez consulter Utilisation de documents Systems Manager Automation dans le Guide de l'utilisateur AWS Toolkit for Visual Studio Code .

Développer le contenu d'un runbook

Une fois votre cas d'utilisation identifié et votre environnement configuré, vous pouvez développer le contenu de votre Runbook. Votre cas d'utilisation et vos préférences dicteront en grande partie les actions d'automatisation ou les runbooks à utiliser dans le contenu de votre runbook. Certaines actions ne prennent en charge qu'un sous-ensemble de paramètres d'entrée, alors que d'autres vous permettent d'accomplir une tâche similaire. D'autres actions ont des sorties spécifiques, telles que `aws:createImage`, alors que certaines actions vous permettent de définir vos propres sorties, telles que `aws:executeAwsApi`.

Si vous ne savez pas comment utiliser une action particulière dans votre runbook, nous vous recommandons d'examiner l'entrée correspondante pour l'action dans Référence sur les actions Systems Manager Automation. Nous vous recommandons également d'examiner le contenu des runbooks prédéfinis afin de disposer d'exemples concrets sur la façon dont ces actions sont utilisées. Pour obtenir d'autres exemples d'applications réelles de runbooks, veuillez consulter Exemples supplémentaires de runbook.

Les didacticiels suivants vous démontreront les différences de simplicité et de flexibilité offertes par le contenu du runbook à l'aide d'exemples sur la façon de corriger des groupes d'instances Amazon EC2 par étapes :

- the section called "Exemple 1 : création de runbooks parent-enfant" - Dans cet exemple, deux runbooks sont utilisés dans une relation parent-enfant. Le runbook parent initie une automatisation de contrôle de débit du runbook enfant.
- the section called "Exemple 2 : runbook scripté" - Cet exemple vous explique comment accomplir les tâches de l'exemple 1 en condensant le contenu dans un seul runbook et en utilisant des scripts dans votre runbook.

Exemple 1 : création de runbooks parent-enfant

L'exemple suivant montre comment créer deux runbooks qui corrigent les groupes étiquetés d'instances Amazon Elastic Compute Cloud (Amazon EC2) par étapes. Ces runbooks sont utilisés dans une relation parent-enfant, le runbook parent initiant une automatisation de contrôle de débit du runbook enfant. Pour de plus amples informations sur les automatisations de contrôle de débit, veuillez consulter Exécution des automatisations à grande échelle. Pour de plus amples informations sur les actions d'automatisation utilisées dans cet exemple, veuillez consulter Référence sur les actions Systems Manager Automation.

Créer le runbook enfant

Cet exemple de runbook contient le scénario suivant. Emily est ingénieur systèmes chez AnyCompany Consultants, LLC. Elle doit configurer l'application de correctifs pour les groupes d'instances Amazon Elastic Compute Cloud (Amazon EC2) qui hébergent des bases de données principale et secondaire. Les applications accèdent à ces bases de données 24 heures sur 24, de sorte que l'une des instances de base de données doit toujours être disponible.

Elle décide que la meilleure approche consiste à appliquer des correctifs par étapes pour les instances. Le groupe principal d'instances de base de données sera donc corrigé en premier, suivi du groupe secondaire d'instances de base de données. En outre, afin de ne pas encourir des coûts supplémentaires en laissant s'exécuter des instances précédemment arrêtées, Emily souhaite que les instances corrigées retrouvent leur statut d'origine avant l'application du correctif.

Emily identifie les groupes principal et secondaire d'instances de base de données grâce aux balises associées aux instances. Elle décide de créer un runbook parent qui démarre une automatisation de contrôle de débit d'un runbook enfant. Ce faisant, elle peut cibler les balises associées aux groupes principal et secondaire d'instances de base de données, et gérer la concurrence des automatisations

enfants. Après avoir examiné les documents Systems Manager (SSM) disponibles pour l'application de correctifs, elle choisit le document `AWS-RunPatchBaseline`. En utilisant ce document SSM, ses collègues peuvent examiner les informations de conformité des correctifs associées, une fois l'opération d'application de correctifs terminée.

Pour commencer à créer le contenu de son runbook, Emily examine les actions d'automatisation disponibles et commence à créer le contenu du runbook enfant comme suit :

1. Tout d'abord, elle fournit des valeurs pour le schéma et la description du runbook, et elle définit les paramètres d'entrée pour le runbook enfant.

   En utilisant le paramètre `AutomationAssumeRole`, Emily et ses collègues peuvent utiliser un rôle IAM existant qui autorise Automation à effectuer des actions dans le runbook en leur nom. Emily utilise le paramètre `InstanceId` pour déterminer l'instance qui doit être corrigée. Le cas échéant, les paramètres `Operation`, `RebootOption` et `SnapshotId` peuvent servir à fournir des valeurs pour documenter les paramètres de `AWS-RunPatchBaseline`. Afin d'empêcher que des valeurs non valides soient fournies à ces paramètres de document, elle définit les `allowedValues` selon besoin.

   YAML

```
schemaVersion: '0.3'
description: 'An example of an Automation runbook that patches groups of Amazon
 EC2 instances in stages.'
assumeRole: '{{AutomationAssumeRole}}'
parameters:
  AutomationAssumeRole:
    type: String
    description: >-
      '(Optional) The Amazon Resource Name (ARN) of the IAM role that allows
 Automation to perform the
      actions on your behalf. If no role is specified, Systems Manager
      Automation uses your IAM permissions to operate this runbook.'
    default: ''
  InstanceId:
    type: String
    description: >-
      '(Required) The instance you want to patch.'
  SnapshotId:
    type: String
    description: '(Optional) The snapshot ID to use to retrieve a patch baseline
 snapshot.'
```

```
    default: ''
  RebootOption:
    type: String
    description: '(Optional) Reboot behavior after a patch Install operation. If
 you choose NoReboot and patches are installed, the instance is marked as non-
compliant until a subsequent reboot and scan.'
    allowedValues:
      - NoReboot
      - RebootIfNeeded
    default: RebootIfNeeded
  Operation:
    type: String
    description: '(Optional) The update or configuration to perform on the
 instance. The system checks if patches specified in the patch baseline are
 installed on the instance. The install operation installs patches missing from
 the baseline.'
    allowedValues:
      - Install
      - Scan
    default: Install
```

JSON

```
{
   "schemaVersion":"0.3",
   "description":"An example of an Automation runbook that patches groups of
 Amazon EC2 instances in stages.",
   "assumeRole":"{{AutomationAssumeRole}}",
   "parameters":{
      "AutomationAssumeRole":{
         "type":"String",
         "description":"(Optional) The Amazon Resource Name (ARN) of the IAM role
 that allows Automation to perform the actions on your behalf. If no role is
 specified, Systems Manager Automation uses your IAM permissions to operate this
 runbook.",
         "default":""
      },
      "InstanceId":{
         "type":"String",
         "description":"(Required) The instance you want to patch."
      },
      "SnapshotId":{
         "type":"String",
```

```
        "description":"(Optional) The snapshot ID to use to retrieve a patch
 baseline snapshot.",
        "default":""
     },
     "RebootOption":{
        "type":"String",
        "description":"(Optional) Reboot behavior after a patch Install
 operation. If you choose NoReboot and patches are installed, the instance is
 marked as non-compliant until a subsequent reboot and scan.",
        "allowedValues":[
           "NoReboot",
           "RebootIfNeeded"
        ],
        "default":"RebootIfNeeded"
     },
     "Operation":{
        "type":"String",
        "description":"(Optional) The update or configuration to perform on
 the instance. The system checks if patches specified in the patch baseline are
 installed on the instance. The install operation installs patches missing from
 the baseline.",
        "allowedValues":[
           "Install",
           "Scan"
        ],
        "default":"Install"
     }
   }
},
```

2. Avec les éléments de niveau supérieur définis, Emily procède à la création des actions qui constituent les `mainSteps` du runbook. La première étape affiche le statut actuel de l'instance cible spécifiée dans le paramètre d'entrée `InstanceId` via l'action `aws:executeAwsApi`. La sortie de cette action sera utilisée dans des actions ultérieures.

   YAML

```
mainSteps:
  - name: getInstanceState
    action: 'aws:executeAwsApi'
    onFailure: Abort
    inputs:
      inputs:
      Service: ec2
```

```
      Api: DescribeInstances
      InstanceIds:
        - '{{InstanceId}}'
    outputs:
      - Name: instanceState
        Selector: '$.Reservations[0].Instances[0].State.Name'
        Type: String
    nextStep: branchOnInstanceState
```

JSON

```
"mainSteps":[
      {
         "name":"getInstanceState",
         "action":"aws:executeAwsApi",
         "onFailure":"Abort",
         "inputs":{
            "inputs":null,
            "Service":"ec2",
            "Api":"DescribeInstances",
            "InstanceIds":[
               "{{InstanceId}}"
            ]
         },
         "outputs":[
            {
               "Name":"instanceState",
               "Selector":"$.Reservations[0].Instances[0].State.Name",
               "Type":"String"
            }
         ],
         "nextStep":"branchOnInstanceState"
      },
```

3. Au lieu de démarrer manuellement et de garder une trace du statut d'origine de chaque instance à corriger, Emily utilise la sortie de l'action précédente pour activer l'automatisation en fonction du statut de l'instance cible. Cela permet à l'automatisation d'exécuter différentes étapes en fonction des conditions définies dans l'action `aws:branch` et de voir son efficacité globale améliorée, sans intervention manuelle.

   Si l'instance a déjà le statut `running`, l'automatisation procède à l'application de correctifs à l'instance avec le document `AWS-RunPatchBaseline` via l'action `aws:runCommand`.

Si l'instance a déjà le statut `stopping`, l'automatisation interroge l'instance afin d'atteindre le statut `stopped` via l'action `aws:waitForAwsResourceProperty`, démarre l'instance via l'action `executeAwsApi` et interroge l'instance afin d'atteindre le statut `running` avant d'appliquer un correctif à l'instance.

Si l'état de l'instance est `stopped`, l'automatisation démarre l'instance et interroge l'instance pour qu'elle atteigne un état `running` avant de corriger l'instance à l'aide des mêmes actions.

YAML

```
- name: branchOnInstanceState
    action: 'aws:branch'
    onFailure: Abort
    inputs:
      Choices:
        - NextStep: startInstance
           Variable: '{{getInstanceState.instanceState}}'
           StringEquals: stopped
         - NextStep: verifyInstanceStopped
           Variable: '{{getInstanceState.instanceState}}'
           StringEquals: stopping
         - NextStep: patchInstance
           Variable: '{{getInstanceState.instanceState}}'
           StringEquals: running
    isEnd: true
  - name: startInstance
    action: 'aws:executeAwsApi'
    onFailure: Abort
    inputs:
      Service: ec2
      Api: StartInstances
      InstanceIds:
        - '{{InstanceId}}'
    nextStep: verifyInstanceRunning
  - name: verifyInstanceRunning
    action: 'aws:waitForAwsResourceProperty'
    timeoutSeconds: 120
    inputs:
      Service: ec2
      Api: DescribeInstances
      InstanceIds:
        - '{{InstanceId}}'
```

```
      PropertySelector: '$.Reservations[0].Instances[0].State.Name'
      DesiredValues:
        - running
    nextStep: patchInstance
  - name: verifyInstanceStopped
    action: 'aws:waitForAwsResourceProperty'
    timeoutSeconds: 120
    inputs:
      Service: ec2
      Api: DescribeInstances
      InstanceIds:
        - '{{InstanceId}}'
      PropertySelector: '$.Reservations[0].Instances[0].State.Name'
      DesiredValues:
        - stopped
      nextStep: startInstance
  - name: patchInstance
    action: 'aws:runCommand'
    onFailure: Abort
    timeoutSeconds: 5400
    inputs:
      DocumentName: 'AWS-RunPatchBaseline'
      InstanceIds:
      - '{{InstanceId}}'
      Parameters:
        SnapshotId: '{{SnapshotId}}'
        RebootOption: '{{RebootOption}}'
        Operation: '{{Operation}}'
```

JSON

```
{
        "name":"branchOnInstanceState",
        "action":"aws:branch",
        "onFailure":"Abort",
        "inputs":{
           "Choices":[
              {
                 "NextStep":"startInstance",
                 "Variable":"{{getInstanceState.instanceState}}",
                 "StringEquals":"stopped"
              },
              {
```

```
                "Or":[
                   {
                      "Variable":"{{getInstanceState.instanceState}}",
                      "StringEquals":"stopping"
                   }
                ],
                "NextStep":"verifyInstanceStopped"
             },
             {
                "NextStep":"patchInstance",
                "Variable":"{{getInstanceState.instanceState}}",
                "StringEquals":"running"
             }
          ]
       },
       "isEnd":true
    },
    {
       "name":"startInstance",
       "action":"aws:executeAwsApi",
       "onFailure":"Abort",
       "inputs":{
          "Service":"ec2",
          "Api":"StartInstances",
          "InstanceIds":[
             "{{InstanceId}}"
          ]
       },
       "nextStep":"verifyInstanceRunning"
    },
    {
       "name":"verifyInstanceRunning",
       "action":"aws:waitForAwsResourceProperty",
       "timeoutSeconds":120,
       "inputs":{
          "Service":"ec2",
          "Api":"DescribeInstances",
          "InstanceIds":[
             "{{InstanceId}}"
          ],
          "PropertySelector":"$.Reservations[0].Instances[0].State.Name",
          "DesiredValues":[
             "running"
          ]
```

```
        },
        "nextStep":"patchInstance"
      },
      {
        "name":"verifyInstanceStopped",
        "action":"aws:waitForAwsResourceProperty",
        "timeoutSeconds":120,
        "inputs":{
          "Service":"ec2",
          "Api":"DescribeInstances",
          "InstanceIds":[
            "{{InstanceId}}"
          ],
          "PropertySelector":"$.Reservations[0].Instances[0].State.Name",
          "DesiredValues":[
            "stopped"
          ],
          "nextStep":"startInstance"
        }
      },
      {
        "name":"patchInstance",
        "action":"aws:runCommand",
        "onFailure":"Abort",
        "timeoutSeconds":5400,
        "inputs":{
          "DocumentName":"AWS-RunPatchBaseline",
          "InstanceIds":[
            "{{InstanceId}}"
          ],
          "Parameters":{
            "SnapshotId":"{{SnapshotId}}",
            "RebootOption":"{{RebootOption}}",
            "Operation":"{{Operation}}"
          }
        }
      },
```

4. Une fois l'opération d'application de correctifs terminée, Emily souhaite que l'automatisation ramène l'instance cible au statut qui était le sien avant le démarrage de l'automatisation. Pour cela, elle utilise à nouveau la sortie de la première action. L'automatisation s'active en fonction du statut d'origine de l'instance cible via l'action `aws:branch`. Si, auparavant, l'instance était

dans un autre statut que `running`, l'instance est arrêtée. Sinon, si l'instance a le statut `running`, l'automatisation se termine.

YAML

```
- name: branchOnOriginalInstanceState
    action: 'aws:branch'
    onFailure: Abort
    inputs:
      Choices:
        - NextStep: stopInstance
          Not:
            Variable: '{{getInstanceState.instanceState}}'
            StringEquals: running
    isEnd: true
  - name: stopInstance
    action: 'aws:executeAwsApi'
    onFailure: Abort
    inputs:
      Service: ec2
      Api: StopInstances
      InstanceIds:
        - '{{InstanceId}}'
```

JSON

```
{
         "name":"branchOnOriginalInstanceState",
         "action":"aws:branch",
         "onFailure":"Abort",
         "inputs":{
            "Choices":[
               {
                  "NextStep":"stopInstance",
                  "Not":{
                     "Variable":"{{getInstanceState.instanceState}}",
                     "StringEquals":"running"
                  }
               }
            ]
         },
         "isEnd":true
      },
```

```
      {
         "name":"stopInstance",
         "action":"aws:executeAwsApi",
         "onFailure":"Abort",
         "inputs":{
            "Service":"ec2",
            "Api":"StopInstances",
            "InstanceIds":[
               "{{InstanceId}}"
            ]
         }
      }
   ]
}
```

5. Emily examine le contenu du runbook enfant terminé et crée le runbook dans le même Compte AWS et la même Région AWS que les instances cibles. Maintenant, elle est prête à passer à la création du contenu du runbook parent. Voici en quoi consiste le contenu du runbook enfant terminé.

   YAML

```
schemaVersion: '0.3'
description: 'An example of an Automation runbook that patches groups of Amazon
 EC2 instances in stages.'
assumeRole: '{{AutomationAssumeRole}}'
parameters:
  AutomationAssumeRole:
    type: String
    description: >-
      '(Optional) The Amazon Resource Name (ARN) of the IAM role that allows
 Automation to perform the
      actions on your behalf. If no role is specified, Systems Manager
      Automation uses your IAM permissions to operate this runbook.'
    default: ''
  InstanceId:
    type: String
    description: >-
      '(Required) The instance you want to patch.'
  SnapshotId:
    type: String
    description: '(Optional) The snapshot ID to use to retrieve a patch baseline
 snapshot.'
    default: ''
```

```
  RebootOption:
    type: String
    description: '(Optional) Reboot behavior after a patch Install operation. If
 you choose NoReboot and patches are installed, the instance is marked as non-
compliant until a subsequent reboot and scan.'
    allowedValues:
      - NoReboot
      - RebootIfNeeded
    default: RebootIfNeeded
  Operation:
    type: String
    description: '(Optional) The update or configuration to perform on the
 instance. The system checks if patches specified in the patch baseline are
 installed on the instance. The install operation installs patches missing from
 the baseline.'
    allowedValues:
      - Install
      - Scan
    default: Install
mainSteps:
  - name: getInstanceState
    action: 'aws:executeAwsApi'
    onFailure: Abort
    inputs:
      inputs:
      Service: ec2
      Api: DescribeInstances
      InstanceIds:
        - '{{InstanceId}}'
    outputs:
      - Name: instanceState
        Selector: '$.Reservations[0].Instances[0].State.Name'
        Type: String
    nextStep: branchOnInstanceState
  - name: branchOnInstanceState
    action: 'aws:branch'
    onFailure: Abort
    inputs:
      Choices:
        - NextStep: startInstance
          Variable: '{{getInstanceState.instanceState}}'
          StringEquals: stopped
        - Or:
            - Variable: '{{getInstanceState.instanceState}}'
```

```
            StringEquals: stopping
          NextStep: verifyInstanceStopped
        - NextStep: patchInstance
          Variable: '{{getInstanceState.instanceState}}'
          StringEquals: running
    isEnd: true
  - name: startInstance
    action: 'aws:executeAwsApi'
    onFailure: Abort
    inputs:
      Service: ec2
      Api: StartInstances
      InstanceIds:
        - '{{InstanceId}}'
    nextStep: verifyInstanceRunning
  - name: verifyInstanceRunning
    action: 'aws:waitForAwsResourceProperty'
    timeoutSeconds: 120
    inputs:
      Service: ec2
      Api: DescribeInstances
      InstanceIds:
        - '{{InstanceId}}'
      PropertySelector: '$.Reservations[0].Instances[0].State.Name'
      DesiredValues:
        - running
    nextStep: patchInstance
  - name: verifyInstanceStopped
    action: 'aws:waitForAwsResourceProperty'
    timeoutSeconds: 120
    inputs:
      Service: ec2
      Api: DescribeInstances
      InstanceIds:
        - '{{InstanceId}}'
      PropertySelector: '$.Reservations[0].Instances[0].State.Name'
      DesiredValues:
        - stopped
      nextStep: startInstance
  - name: patchInstance
    action: 'aws:runCommand'
    onFailure: Abort
    timeoutSeconds: 5400
    inputs:
```

```
      DocumentName: 'AWS-RunPatchBaseline'
      InstanceIds:
      - '{{InstanceId}}'
      Parameters:
        SnapshotId: '{{SnapshotId}}'
        RebootOption: '{{RebootOption}}'
        Operation: '{{Operation}}'
  - name: branchOnOriginalInstanceState
    action: 'aws:branch'
    onFailure: Abort
    inputs:
      Choices:
        - NextStep: stopInstance
          Not:
            Variable: '{{getInstanceState.instanceState}}'
            StringEquals: running
    isEnd: true
  - name: stopInstance
    action: 'aws:executeAwsApi'
    onFailure: Abort
    inputs:
      Service: ec2
      Api: StopInstances
      InstanceIds:
        - '{{InstanceId}}'
```

JSON

```
{
   "schemaVersion":"0.3",
   "description":"An example of an Automation runbook that patches groups of
 Amazon EC2 instances in stages.",
   "assumeRole":"{{AutomationAssumeRole}}",
   "parameters":{
      "AutomationAssumeRole":{
         "type":"String",
         "description":"'(Optional) The Amazon Resource Name (ARN) of the IAM
 role that allows Automation to perform the actions on your behalf. If no role is
 specified, Systems Manager Automation uses your IAM permissions to operate this
 runbook.'",
         "default":""
      },
      "InstanceId":{
```

```
        "type":"String",
        "description":"'(Required) The instance you want to patch.'"
     },
     "SnapshotId":{
        "type":"String",
        "description":"(Optional) The snapshot ID to use to retrieve a patch
baseline snapshot.",
        "default":""
     },
     "RebootOption":{
        "type":"String",
        "description":"(Optional) Reboot behavior after a patch Install
operation. If you choose NoReboot and patches are installed, the instance is
marked as non-compliant until a subsequent reboot and scan.",
        "allowedValues":[
           "NoReboot",
           "RebootIfNeeded"
        ],
        "default":"RebootIfNeeded"
     },
     "Operation":{
        "type":"String",
        "description":"(Optional) The update or configuration to perform on
the instance. The system checks if patches specified in the patch baseline are
installed on the instance. The install operation installs patches missing from
the baseline.",
        "allowedValues":[
           "Install",
           "Scan"
        ],
        "default":"Install"
     }
  },
  "mainSteps":[
     {
        "name":"getInstanceState",
        "action":"aws:executeAwsApi",
        "onFailure":"Abort",
        "inputs":{
           "inputs":null,
           "Service":"ec2",
           "Api":"DescribeInstances",
           "InstanceIds":[
              "{{InstanceId}}"
```

```
            ]
         },
         "outputs":[
            {
               "Name":"instanceState",
               "Selector":"$.Reservations[0].Instances[0].State.Name",
               "Type":"String"
            }
         ],
         "nextStep":"branchOnInstanceState"
      },
      {
         "name":"branchOnInstanceState",
         "action":"aws:branch",
         "onFailure":"Abort",
         "inputs":{
            "Choices":[
               {
                  "NextStep":"startInstance",
                  "Variable":"{{getInstanceState.instanceState}}",
                  "StringEquals":"stopped"
               },
               {
                  "Or":[
                     {
                        "Variable":"{{getInstanceState.instanceState}}",
                        "StringEquals":"stopping"
                     }
                  ],
                  "NextStep":"verifyInstanceStopped"
               },
               {
                  "NextStep":"patchInstance",
                  "Variable":"{{getInstanceState.instanceState}}",
                  "StringEquals":"running"
               }
            ]
         },
         "isEnd":true
      },
      {
         "name":"startInstance",
         "action":"aws:executeAwsApi",
         "onFailure":"Abort",
```

```
         "inputs":{
            "Service":"ec2",
            "Api":"StartInstances",
            "InstanceIds":[
               "{{InstanceId}}"
            ]
         },
         "nextStep":"verifyInstanceRunning"
      },
      {
         "name":"verifyInstanceRunning",
         "action":"aws:waitForAwsResourceProperty",
         "timeoutSeconds":120,
         "inputs":{
            "Service":"ec2",
            "Api":"DescribeInstances",
            "InstanceIds":[
               "{{InstanceId}}"
            ],
            "PropertySelector":"$.Reservations[0].Instances[0].State.Name",
            "DesiredValues":[
               "running"
            ]
         },
         "nextStep":"patchInstance"
      },
      {
         "name":"verifyInstanceStopped",
         "action":"aws:waitForAwsResourceProperty",
         "timeoutSeconds":120,
         "inputs":{
            "Service":"ec2",
            "Api":"DescribeInstances",
            "InstanceIds":[
               "{{InstanceId}}"
            ],
            "PropertySelector":"$.Reservations[0].Instances[0].State.Name",
            "DesiredValues":[
               "stopped"
            ],
            "nextStep":"startInstance"
         }
      },
      {
```

```
            "name":"patchInstance",
            "action":"aws:runCommand",
            "onFailure":"Abort",
            "timeoutSeconds":5400,
            "inputs":{
               "DocumentName":"AWS-RunPatchBaseline",
               "InstanceIds":[
                  "{{InstanceId}}"
               ],
               "Parameters":{
                  "SnapshotId":"{{SnapshotId}}",
                  "RebootOption":"{{RebootOption}}",
                  "Operation":"{{Operation}}"
               }
            }
         },
         {
            "name":"branchOnOriginalInstanceState",
            "action":"aws:branch",
            "onFailure":"Abort",
            "inputs":{
               "Choices":[
                  {
                     "NextStep":"stopInstance",
                     "Not":{
                        "Variable":"{{getInstanceState.instanceState}}",
                        "StringEquals":"running"
                     }
                  }
               ]
            },
            "isEnd":true
         },
         {
            "name":"stopInstance",
            "action":"aws:executeAwsApi",
            "onFailure":"Abort",
            "inputs":{
               "Service":"ec2",
               "Api":"StopInstances",
               "InstanceIds":[
                  "{{InstanceId}}"
               ]
            }
```

```
      }
   ]
}
```

Pour de plus amples informations sur les actions d'automatisation utilisées dans cet exemple, veuillez consulter Référence sur les actions Systems Manager Automation.

Créer le runbook parent

Cet exemple de runbook poursuit le scénario décrit dans la section précédente. Maintenant qu'Emily a créé le runbook enfant, elle commence à créer le contenu du runbook parent comme suit :

1. Tout d'abord, elle fournit des valeurs pour le schéma et la description du runbook, et elle définit les paramètres d'entrée pour le runbook parent.

   En utilisant le paramètre `AutomationAssumeRole`, Emily et ses collègues peuvent utiliser un rôle IAM existant qui autorise Automation à effectuer des actions dans le runbook en leur nom. Emily utilise les paramètres `PatchGroupPrimaryKey` et `PatchGroupPrimaryValue` pour spécifier la balise associée au groupe principal d'instances de base de données qui seront corrigées. Elle utilise les paramètres `PatchGroupSecondaryKey` et `PatchGroupSecondaryValue` pour spécifier la balise associée au groupe secondaire d'instances de base de données qui seront corrigées.

   YAML

```
description: 'An example of an Automation runbook that patches groups of Amazon
 EC2 instances in stages.'
schemaVersion: '0.3'
assumeRole: '{{AutomationAssumeRole}}'
parameters:
  AutomationAssumeRole:
    type: String
    description: '(Optional) The Amazon Resource Name (ARN) of the IAM role that
 allows Automation to perform the actions on your behalf. If no role is specified,
 Systems Manager Automation uses your IAM permissions to operate this runbook.'
    default: ''
  PatchGroupPrimaryKey:
    type: String
    description: '(Required) The key of the tag for the primary group of instances
 you want to patch.''
  PatchGroupPrimaryValue:
```

```
    type: String
    description: '(Required) The value of the tag for the primary group of
instances you want to patch.'
 PatchGroupSecondaryKey:
    type: String
    description: '(Required) The key of the tag for the secondary group of
instances you want to patch.'
 PatchGroupSecondaryValue:
    type: String
    description: '(Required) The value of the tag for the secondary group of
instances you want to patch.'
```

JSON

```
{
   "schemaVersion": "0.3",
   "description": "An example of an Automation runbook that patches groups of
 Amazon EC2 instances in stages.",
   "assumeRole": "{{AutomationAssumeRole}}",
   "parameters": {
      "AutomationAssumeRole": {
         "type": "String",
         "description": "(Optional) The Amazon Resource Name (ARN) of the IAM
 role that allows Automation to perform the actions on your behalf. If no role is
 specified, Systems Manager Automation uses your IAM permissions to operate this
 runbook.",
         "default": ""
      },
      "PatchGroupPrimaryKey": {
         "type": "String",
         "description": "(Required) The key of the tag for the primary group of
instances you want to patch."
      },
      "PatchGroupPrimaryValue": {
         "type": "String",
         "description": "(Required) The value of the tag for the primary group of
instances you want to patch."
      },
      "PatchGroupSecondaryKey": {
         "type": "String",
         "description": "(Required) The key of the tag for the secondary group of
instances you want to patch."
      },
```

```
      "PatchGroupSecondaryValue": {
         "type": "String",
         "description": "(Required) The value of the tag for the secondary group
 of instances you want to patch."
      }
   }
},
```

2. Avec les éléments de niveau supérieur définis, Emily procède à la création des actions qui constituent les `mainSteps` du runbook.

   La première action démarre une automatisation de contrôle de débit à l'aide du runbook enfant qu'elle vient de créer et qui cible les instances associées à la balise spécifiée dans les paramètres d'entrée `PatchGroupPrimaryKey` et `PatchGroupPrimaryValue`. Elle utilise les valeurs fournies aux paramètres d'entrée pour spécifier la clé et la valeur de la balise associée au groupe principal d'instances de base de données qu'elle souhaite corriger.

   Une fois la première automatisation terminée, la seconde action démarre une autre automatisation de contrôle de débit à l'aide du runbook enfant et qui cible les instances associées à la balise spécifiée dans les paramètres d'entrée `PatchGroupSecondaryKey` et `PatchGroupSecondaryValue`. Elle utilise les valeurs fournies aux paramètres d'entrée pour spécifier la clé et la valeur de la balise associée au groupe secondaire d'instances de base de données qu'elle souhaite corriger.

   YAML

```
mainSteps:
  - name: patchPrimaryTargets
    action: 'aws:executeAutomation'
    onFailure: Abort
    timeoutSeconds: 7200
    inputs:
      DocumentName: RunbookTutorialChildAutomation
      Targets:
        - Key: 'tag:{{PatchGroupPrimaryKey}}'
          Values:
            - '{{PatchGroupPrimaryValue}}'
      TargetParameterName: 'InstanceId'
  - name: patchSecondaryTargets
    action: 'aws:executeAutomation'
    onFailure: Abort
    timeoutSeconds: 7200
```

```
    inputs:
      DocumentName: RunbookTutorialChildAutomation
      Targets:
        - Key: 'tag:{{PatchGroupSecondaryKey}}'
          Values:
            - '{{PatchGroupSecondaryValue}}'
      TargetParameterName: 'InstanceId'
```

JSON

```
"mainSteps":[
      {
         "name":"patchPrimaryTargets",
         "action":"aws:executeAutomation",
         "onFailure":"Abort",
         "timeoutSeconds":7200,
         "inputs":{
            "DocumentName":"RunbookTutorialChildAutomation",
            "Targets":[
               {
                  "Key":"tag:{{PatchGroupPrimaryKey}}",
                  "Values":[
                     "{{PatchGroupPrimaryValue}}"
                  ]
               }
            ],
            "TargetParameterName":"InstanceId"
         }
      },
      {
         "name":"patchSecondaryTargets",
         "action":"aws:executeAutomation",
         "onFailure":"Abort",
         "timeoutSeconds":7200,
         "inputs":{
            "DocumentName":"RunbookTutorialChildAutomation",
            "Targets":[
               {
                  "Key":"tag:{{PatchGroupSecondaryKey}}",
                  "Values":[
                     "{{PatchGroupSecondaryValue}}"
                  ]
               }
```

```
            ],
            "TargetParameterName":"InstanceId"
         }
      }
   ]
}
```

3. Emily examine le contenu du runbook parent terminé et crée le runbook dans le même Compte AWS et la même Région AWS que les instances cibles. Maintenant, elle est prête à tester ses runbooks afin de s'assurer que l'automatisation fonctionne correctement avant de les implémenter dans son environnement de production. Voici en quoi consiste le contenu du runbook parent terminé.

   YAML

```
description: An example of an Automation runbook that patches groups of Amazon EC2
 instances in stages.
schemaVersion: '0.3'
assumeRole: '{{AutomationAssumeRole}}'
parameters:
  AutomationAssumeRole:
    type: String
    description: '(Optional) The Amazon Resource Name (ARN) of the IAM role that
 allows Automation to perform the actions on your behalf. If no role is specified,
 Systems Manager Automation uses your IAM permissions to operate this runbook.'
    default: ''
  PatchGroupPrimaryKey:
    type: String
    description: (Required) The key of the tag for the primary group of instances
 you want to patch.
  PatchGroupPrimaryValue:
    type: String
    description: '(Required) The value of the tag for the primary group of
 instances you want to patch. '
  PatchGroupSecondaryKey:
    type: String
    description: (Required) The key of the tag for the secondary group of
 instances you want to patch.
  PatchGroupSecondaryValue:
    type: String
    description: '(Required) The value of the tag for the secondary group of
 instances you want to patch.  '
mainSteps:
```

```
  - name: patchPrimaryTargets
    action: 'aws:executeAutomation'
    onFailure: Abort
    timeoutSeconds: 7200
    inputs:
      DocumentName: RunbookTutorialChildAutomation
      Targets:
        - Key: 'tag:{{PatchGroupPrimaryKey}}'
          Values:
            - '{{PatchGroupPrimaryValue}}'
      TargetParameterName: 'InstanceId'
  - name: patchSecondaryTargets
    action: 'aws:executeAutomation'
    onFailure: Abort
    timeoutSeconds: 7200
    inputs:
      DocumentName: RunbookTutorialChildAutomation
      Targets:
        - Key: 'tag:{{PatchGroupSecondaryKey}}'
          Values:
            - '{{PatchGroupSecondaryValue}}'
      TargetParameterName: 'InstanceId'
```

JSON

```
{
   "description":"An example of an Automation runbook that patches groups of
 Amazon EC2 instances in stages.",
   "schemaVersion":"0.3",
   "assumeRole":"{{AutomationAssumeRole}}",
   "parameters":{
      "AutomationAssumeRole":{
         "type":"String",
         "description":"(Optional) The Amazon Resource Name (ARN) of the IAM role
 that allows Automation to perform the actions on your behalf. If no role is
 specified, Systems Manager Automation uses your IAM permissions to operate this
 runbook.",
         "default":""
      },
      "PatchGroupPrimaryKey":{
         "type":"String",
         "description":"(Required) The key of the tag for the primary group of
 instances you want to patch."
```

```
      },
      "PatchGroupPrimaryValue":{
         "type":"String",
         "description":"(Required) The value of the tag for the primary group of
instances you want to patch. "
      },
      "PatchGroupSecondaryKey":{
         "type":"String",
         "description":"(Required) The key of the tag for the secondary group of
instances you want to patch."
      },
      "PatchGroupSecondaryValue":{
         "type":"String",
         "description":"(Required) The value of the tag for the secondary group of
instances you want to patch.  "
      }
   },
   "mainSteps":[
      {
         "name":"patchPrimaryTargets",
         "action":"aws:executeAutomation",
         "onFailure":"Abort",
         "timeoutSeconds":7200,
         "inputs":{
            "DocumentName":"RunbookTutorialChildAutomation",
            "Targets":[
               {
                  "Key":"tag:{{PatchGroupPrimaryKey}}",
                  "Values":[
                     "{{PatchGroupPrimaryValue}}"
                  ]
               }
            ],
            "TargetParameterName":"InstanceId"
         }
      },
      {
         "name":"patchSecondaryTargets",
         "action":"aws:executeAutomation",
         "onFailure":"Abort",
         "timeoutSeconds":7200,
         "inputs":{
            "DocumentName":"RunbookTutorialChildAutomation",
            "Targets":[
```

```
                {
                   "Key":"tag:{{PatchGroupSecondaryKey}}",
                   "Values":[
                      "{{PatchGroupSecondaryValue}}"
                   ]
                }
             ],
             "TargetParameterName":"InstanceId"
          }
       }
    ]
}
```

Pour de plus amples informations sur les actions d'automatisation utilisées dans cet exemple, veuillez consulter Référence sur les actions Systems Manager Automation.

Exemple 2 : runbook scripté

Cet exemple de runbook contient le scénario suivant. Emily est ingénieur systèmes chez AnyCompany Consultants, LLC. Précédemment, elle a créé deux runbooks qui sont utilisés dans une relation parent-enfant pour corriger des groupes d'instances Amazon Elastic Compute Cloud (Amazon EC2) qui hébergent des bases de données principale et secondaire. Les applications accèdent à ces bases de données 24 heures sur 24, de sorte que l'une des instances de base de données doit toujours être disponible.

Sur la base de cette exigence, elle a créé une solution qui corrige les instances par étapes à l'aide du document `AWS-RunPatchBaseline`Systems Manager (SSM). En utilisant ce document SSM, ses collègues peuvent examiner les informations de conformité des correctifs associées, une fois l'opération d'application de correctifs terminée.

Le groupe principal d'instances de base de données sera donc corrigé en premier, suivi du groupe secondaire d'instances de base de données. De plus, pour éviter d'encourir des coûts supplémentaires en laissant en cours d'exécution des instances qui avaient été précédemment arrêtées, Emily s'est assurée que l'automatisation remettait les instances corrigées à leur état d'origine avant que la correction ne se produise. Emily a utilisé des balises associées aux groupes primaire et secondaire d'instances de base de données afin d'identifier les instances à corriger dans l'ordre voulu.

Sa solution automatisée existante fonctionne, mais elle voudrait l'améliorer, si possible. Pour faciliter la maintenance du contenu du runbook et les efforts de dépannage, elle souhaite condenser

l'automatisation en un runbook unique et réduire le nombre de paramètres d'entrée. En outre, elle voudrait éviter de créer plusieurs automatisations enfants.

Après avoir examiné les actions d'automatisation disponibles, elle détermine qu'elle peut améliorer sa solution en utilisant l'action `aws:executeScript` pour exécuter ses scripts Python personnalisés. Elle commence alors à créer le contenu du runbook comme suit :

1. Tout d'abord, elle fournit des valeurs pour le schéma et la description du runbook, et elle définit les paramètres d'entrée pour le runbook parent.

   En utilisant le paramètre `AutomationAssumeRole`, Emily et ses collègues peuvent utiliser un rôle IAM existant qui autorise Automation à effectuer des actions dans le runbook en leur nom. Contrairement à l'exemple 1, le paramètre `AutomationAssumeRole` est obligatoire, et non plus facultatif. Comme ce runbook inclut des actions `aws:executeScript`, un rôle de service AWS Identity and Access Management (IAM) (ou une opération assume-role) est toujours requis. Cette exigence est nécessaire car certains des scripts Python spécifiés pour les actions appellent des opérations d'API AWS.

   Emily utilise les paramètres `PrimaryPatchGroupTag` et `SecondaryPatchGroupTag` pour spécifier les balises associées au groupe principal et secondaire d'instances de base de données qui seront corrigées. Pour simplifier les paramètres d'entrée requis, elle décide d'utiliser des paramètres `StringMap` plutôt que plusieurs paramètres `String`, comme elle l'a fait dans l'exemple de runbook 1. Le cas échéant, les paramètres `Operation`, `RebootOption` et `SnapshotId` peuvent servir à fournir des valeurs pour documenter les paramètres de `AWS-RunPatchBaseline`. Afin d'empêcher que des valeurs non valides soient fournies à ces paramètres de document, elle définit les `allowedValues` selon besoin.

   YAML

```
description: 'An example of an Automation runbook that patches groups of Amazon
 EC2 instances in stages.'
schemaVersion: '0.3'
assumeRole: '{{AutomationAssumeRole}}'
parameters:
  AutomationAssumeRole:
    type: String
    description: '(Required) The Amazon Resource Name (ARN) of the IAM role that
 allows Automation to perform the actions on your behalf. If no role is specified,
 Systems Manager Automation uses your IAM permissions to operate this runbook.'
  PrimaryPatchGroupTag:
    type: StringMap
```

```
    description: '(Required) The tag for the primary group of instances you want
 to patch. Specify a key-value pair. Example: {"key" : "value"}'
  SecondaryPatchGroupTag:
    type: StringMap
    description: '(Required) The tag for the secondary group of instances you want
 to patch. Specify a key-value pair. Example: {"key" : "value"}'
  SnapshotId:
    type: String
    description: '(Optional) The snapshot ID to use to retrieve a patch baseline
 snapshot.'
    default: ''
  RebootOption:
    type: String
    description: '(Optional) Reboot behavior after a patch Install operation. If
 you choose NoReboot and patches are installed, the instance is marked as non-
compliant until a subsequent reboot and scan.'
    allowedValues:
      - NoReboot
      - RebootIfNeeded
    default: RebootIfNeeded
  Operation:
    type: String
    description: '(Optional) The update or configuration to perform on the
 instance. The system checks if patches specified in the patch baseline are
 installed on the instance. The install operation installs patches missing from
 the baseline.'
    allowedValues:
      - Install
      - Scan
    default: Install
```

JSON

```
{
   "description":"An example of an Automation runbook that patches groups of
 Amazon EC2 instances in stages.",
   "schemaVersion":"0.3",
   "assumeRole":"{{AutomationAssumeRole}}",
   "parameters":{
      "AutomationAssumeRole":{
         "type":"String",
         "description":"(Required) The Amazon Resource Name (ARN) of the IAM role
 that allows Automation to perform the actions on your behalf. If no role is
```

```
specified, Systems Manager Automation uses your IAM permissions to operate this
runbook."
     },
     "PrimaryPatchGroupTag":{
        "type":"StringMap",
        "description":"(Required) The tag for the primary group of instances you
want to patch. Specify a key-value pair. Example: {\"key\" : \"value\"}"
     },
     "SecondaryPatchGroupTag":{
        "type":"StringMap",
        "description":"(Required) The tag for the secondary group of instances
you want to patch. Specify a key-value pair. Example: {\"key\" : \"value\"}"
     },
     "SnapshotId":{
        "type":"String",
        "description":"(Optional) The snapshot ID to use to retrieve a patch
baseline snapshot.",
        "default":""
     },
     "RebootOption":{
        "type":"String",
        "description":"(Optional) Reboot behavior after a patch Install
operation. If you choose NoReboot and patches are installed, the instance is
marked as non-compliant until a subsequent reboot and scan.",
        "allowedValues":[
           "NoReboot",
           "RebootIfNeeded"
        ],
        "default":"RebootIfNeeded"
     },
     "Operation":{
        "type":"String",
        "description":"(Optional) The update or configuration to perform on
the instance. The system checks if patches specified in the patch baseline are
installed on the instance. The install operation installs patches missing from
the baseline.",
        "allowedValues":[
           "Install",
           "Scan"
        ],
        "default":"Install"
     }
  }
```

```
},
```

2. Avec les éléments de niveau supérieur définis, Emily procède à la création des actions qui constituent les `mainSteps` du runbook. La première étape rassemble les ID de toutes les instances associées à la balise spécifiée dans le paramètre `PrimaryPatchGroupTag` et génère un paramètre `StringMap` contenant l'ID d'instance, ainsi que le statut actuel de l'instance. La sortie de cette action sera utilisée dans des actions ultérieures.

   Remarque : le paramètre d'entrée `script` n'est pas pris en charge pour les runbooks JSON. Les runbooks JSON doivent fournir du contenu de script à l'aide du paramètre d'entrée `attachment`.

   YAML

```
mainSteps:
  - name: getPrimaryInstanceState
    action: 'aws:executeScript'
    timeoutSeconds: 120
    onFailure: Abort
    inputs:
      Runtime: python3.7
      Handler: getInstanceStates
      InputPayload:
        primaryTag: '{{PrimaryPatchGroupTag}}'
      Script: |-
        def getInstanceStates(events,context):
          import boto3

          #Initialize client
          ec2 = boto3.client('ec2')
          tag = events['primaryTag']
          tagKey, tagValue = list(tag.items())[0]
          instanceQuery = ec2.describe_instances(
          Filters=[
              {
                  "Name": "tag:" + tagKey,
                  "Values": [tagValue]
              }]
          )
          if not instanceQuery['Reservations']:
              noInstancesForTagString = "No instances found for specified tag."
              return({ 'noInstancesFound' : noInstancesForTagString })
          else:
              queryResponse = instanceQuery['Reservations']
```

```
            originalInstanceStates = {}
            for results in queryResponse:
                instanceSet = results['Instances']
                for instance in instanceSet:
                    instanceId = instance['InstanceId']
                    originalInstanceStates[instanceId] = instance['State']
['Name']
            return originalInstanceStates
    outputs:
      - Name: originalInstanceStates
        Selector: $.Payload
        Type: StringMap
    nextStep: verifyPrimaryInstancesRunning
```

JSON

```
"mainSteps":[
      {
         "name":"getPrimaryInstanceState",
         "action":"aws:executeScript",
         "timeoutSeconds":120,
         "onFailure":"Abort",
         "inputs":{
            "Runtime":"python3.7",
            "Handler":"getInstanceStates",
            "InputPayload":{
               "primaryTag":"{{PrimaryPatchGroupTag}}"
            },
            "Script":"..."
         },
         "outputs":[
            {
               "Name":"originalInstanceStates",
               "Selector":"$.Payload",
               "Type":"StringMap"
            }
         ],
         "nextStep":"verifyPrimaryInstancesRunning"
      },
```

3. Emily utilise la sortie de l'action précédente dans une autre action `aws:executeScript` afin de vérifier que toutes les instances associées à la balise spécifiée dans le paramètre `PrimaryPatchGroupTag` ont un statut `running`.

Si l'instance a déjà le statut `running` ou `shutting-down`, le script continue à boucler sur les instances restantes.

Si l'instance a le statut `stopping`, le script interroge l'instance afin d'atteindre le statut `stopped` et démarre l'instance.

Si l'instance a le statut `stopped`, le script démarre l'instance.

YAML

```
- name: verifyPrimaryInstancesRunning
    action: 'aws:executeScript'
    timeoutSeconds: 600
    onFailure: Abort
    inputs:
      Runtime: python3.7
      Handler: verifyInstancesRunning
      InputPayload:
        targetInstances: '{{getPrimaryInstanceState.originalInstanceStates}}'
      Script: |-
        def verifyInstancesRunning(events,context):
          import boto3

          #Initialize client
          ec2 = boto3.client('ec2')
          instanceDict = events['targetInstances']
          for instance in instanceDict:
            if instanceDict[instance] == 'stopped':
                print("The target instance " + instance + " is stopped. The
 instance will now be started.")
                ec2.start_instances(
                    InstanceIds=[instance]
                    )
            elif instanceDict[instance] == 'stopping':
                print("The target instance " + instance + " is stopping. Polling
 for instance to reach stopped state.")
                while instanceDict[instance] != 'stopped':
                    poll = ec2.get_waiter('instance_stopped')
                    poll.wait(
                        InstanceIds=[instance]
                    )
                ec2.start_instances(
                    InstanceIds=[instance]
```

```
                )
            else:
              pass
    nextStep: waitForPrimaryRunningInstances
```

JSON

```
{
         "name":"verifyPrimaryInstancesRunning",
         "action":"aws:executeScript",
         "timeoutSeconds":600,
         "onFailure":"Abort",
         "inputs":{
            "Runtime":"python3.7",
            "Handler":"verifyInstancesRunning",
            "InputPayload":{

 "targetInstances":"{{getPrimaryInstanceState.originalInstanceStates}}"
            },
            "Script":"..."
         },
         "nextStep":"waitForPrimaryRunningInstances"
      },
```

4. Emily vérifie que toutes les instances associées à la balise spécifiée dans le paramètre `PrimaryPatchGroupTag` ont été démarrées ou ont le statut `running`. Ensuite, elle utilise un autre script pour vérifier que toutes les instances, y compris celles qui ont été démarrées dans l'action précédente, ont atteint le statut `running`.

   YAML

```
- name: waitForPrimaryRunningInstances
    action: 'aws:executeScript'
    timeoutSeconds: 300
    onFailure: Abort
    inputs:
      Runtime: python3.7
      Handler: waitForRunningInstances
      InputPayload:
        targetInstances: '{{getPrimaryInstanceState.originalInstanceStates}}'
      Script: |-
        def waitForRunningInstances(events,context):
          import boto3
```

```
          #Initialize client
          ec2 = boto3.client('ec2')
          instanceDict = events['targetInstances']
          for instance in instanceDict:
              poll = ec2.get_waiter('instance_running')
              poll.wait(
                  InstanceIds=[instance]
              )
    nextStep: returnPrimaryTagKey
```

JSON

```
{
         "name":"waitForPrimaryRunningInstances",
         "action":"aws:executeScript",
         "timeoutSeconds":300,
         "onFailure":"Abort",
         "inputs":{
            "Runtime":"python3.7",
            "Handler":"waitForRunningInstances",
            "InputPayload":{

 "targetInstances":"{{getPrimaryInstanceState.originalInstanceStates}}"
            },
            "Script":"..."
         },
         "nextStep":"returnPrimaryTagKey"
      },
```

5. Emily utilise deux scripts supplémentaires pour renvoyer les valeurs `String` individuelles de la clé et de la valeur de la balise spécifiée dans le paramètre `PrimaryPatchGroupTag`. Les valeurs renvoyées par ces actions lui permettent de fournir des valeurs directement au paramètre `Targets` pour le document `AWS-RunPatchBaseline`. L'automatisation procède à l'application de correctifs à l'instance avec le document `AWS-RunPatchBaseline` via l'action `aws:runCommand`.

   YAML

```
- name: returnPrimaryTagKey
    action: 'aws:executeScript'
    timeoutSeconds: 120
    onFailure: Abort
```

```
    inputs:
      Runtime: python3.7
      Handler: returnTagValues
      InputPayload:
        primaryTag: '{{PrimaryPatchGroupTag}}'
      Script: |-
        def returnTagValues(events,context):
          tag = events['primaryTag']
          tagKey = list(tag)[0]
          stringKey = "tag:" + tagKey
          return {'tagKey' : stringKey}
    outputs:
      - Name: Payload
        Selector: $.Payload
        Type: StringMap
      - Name: primaryPatchGroupKey
        Selector: $.Payload.tagKey
        Type: String
    nextStep: returnPrimaryTagValue
  - name: returnPrimaryTagValue
    action: 'aws:executeScript'
    timeoutSeconds: 120
    onFailure: Abort
    inputs:
      Runtime: python3.7
      Handler: returnTagValues
      InputPayload:
        primaryTag: '{{PrimaryPatchGroupTag}}'
      Script: |-
        def returnTagValues(events,context):
          tag = events['primaryTag']
          tagKey = list(tag)[0]
          tagValue = tag[tagKey]
          return {'tagValue' : tagValue}
    outputs:
      - Name: Payload
        Selector: $.Payload
        Type: StringMap
      - Name: primaryPatchGroupValue
        Selector: $.Payload.tagValue
        Type: String
    nextStep: patchPrimaryInstances
  - name: patchPrimaryInstances
    action: 'aws:runCommand'
```

```
    onFailure: Abort
    timeoutSeconds: 7200
    inputs:
      DocumentName: AWS-RunPatchBaseline
      Parameters:
        SnapshotId: '{{SnapshotId}}'
        RebootOption: '{{RebootOption}}'
        Operation: '{{Operation}}'
      Targets:
        - Key: '{{returnPrimaryTagKey.primaryPatchGroupKey}}'
          Values:
            - '{{returnPrimaryTagValue.primaryPatchGroupValue}}'
      MaxConcurrency: 10%
      MaxErrors: 10%
    nextStep: returnPrimaryToOriginalState
```

JSON

```
{
         "name":"returnPrimaryTagKey",
         "action":"aws:executeScript",
         "timeoutSeconds":120,
         "onFailure":"Abort",
         "inputs":{
            "Runtime":"python3.7",
            "Handler":"returnTagValues",
            "InputPayload":{
               "primaryTag":"{{PrimaryPatchGroupTag}}"
            },
            "Script":"..."
         },
         "outputs":[
            {
               "Name":"Payload",
               "Selector":"$.Payload",
               "Type":"StringMap"
            },
            {
               "Name":"primaryPatchGroupKey",
               "Selector":"$.Payload.tagKey",
               "Type":"String"
            }
         ],
```

```
            "nextStep":"returnPrimaryTagValue"
         },
         {
            "name":"returnPrimaryTagValue",
            "action":"aws:executeScript",
            "timeoutSeconds":120,
            "onFailure":"Abort",
            "inputs":{
               "Runtime":"python3.7",
               "Handler":"returnTagValues",
               "InputPayload":{
                  "primaryTag":"{{PrimaryPatchGroupTag}}"
               },
               "Script":"..."
            },
            "outputs":[
               {
                  "Name":"Payload",
                  "Selector":"$.Payload",
                  "Type":"StringMap"
               },
               {
                  "Name":"primaryPatchGroupValue",
                  "Selector":"$.Payload.tagValue",
                  "Type":"String"
               }
            ],
            "nextStep":"patchPrimaryInstances"
         },
         {
            "name":"patchPrimaryInstances",
            "action":"aws:runCommand",
            "onFailure":"Abort",
            "timeoutSeconds":7200,
            "inputs":{
               "DocumentName":"AWS-RunPatchBaseline",
               "Parameters":{
                  "SnapshotId":"{{SnapshotId}}",
                  "RebootOption":"{{RebootOption}}",
                  "Operation":"{{Operation}}"
               },
               "Targets":[
                  {
                     "Key":"{{returnPrimaryTagKey.primaryPatchGroupKey}}",
```

```
                "Values":[
                   "{{returnPrimaryTagValue.primaryPatchGroupValue}}"
                ]
             }
          ],
          "MaxConcurrency":"10%",
          "MaxErrors":"10%"
       },
       "nextStep":"returnPrimaryToOriginalState"
    },
```

6. Une fois l'opération d'application de correctifs terminée, Emily souhaite que l'automatisation ramène les instances cibles associées à la balise spécifiée dans le paramètre `PrimaryPatchGroupTag` au statut qui était le leur avant le démarrage de l'automatisation. Pour cela, elle utilise à nouveau la sortie de la première action dans un script. Sur la base du statut d'origine de l'instance cible, si l'instance était auparavant dans un autre statut que `running`, l'instance est arrêtée. Autrement, si l'instance a le statut `running`, le script continue à boucler sur les instances restantes.

   YAML

```
- name: returnPrimaryToOriginalState
    action: 'aws:executeScript'
    timeoutSeconds: 600
    onFailure: Abort
    inputs:
      Runtime: python3.7
      Handler: returnToOriginalState
      InputPayload:
        targetInstances: '{{getPrimaryInstanceState.originalInstanceStates}}'
      Script: |-
        def returnToOriginalState(events,context):
          import boto3

          #Initialize client
          ec2 = boto3.client('ec2')
          instanceDict = events['targetInstances']
          for instance in instanceDict:
            if instanceDict[instance] == 'stopped' or instanceDict[instance] ==
 'stopping':
                ec2.stop_instances(
                    InstanceIds=[instance]
                    )
```

```
            else:
              pass
    nextStep: getSecondaryInstanceState
```

JSON

```
{
         "name":"returnPrimaryToOriginalState",
         "action":"aws:executeScript",
         "timeoutSeconds":600,
         "onFailure":"Abort",
         "inputs":{
            "Runtime":"python3.7",
            "Handler":"returnToOriginalState",
            "InputPayload":{

 "targetInstances":"{{getPrimaryInstanceState.originalInstanceStates}}"
            },
            "Script":"..."
         },
         "nextStep":"getSecondaryInstanceState"
      },
```

7. L'opération d'application de correctifs est terminée pour les instances associées à la balise spécifiée dans le paramètre `PrimaryPatchGroupTag`. Maintenant, Emily duplique toutes les actions précédentes dans son contenu de runbook afin de cibler les instances associées à la balise spécifiée dans le paramètre `SecondaryPatchGroupTag`.

   YAML

```
- name: getSecondaryInstanceState
    action: 'aws:executeScript'
    timeoutSeconds: 120
    onFailure: Abort
    inputs:
      Runtime: python3.7
      Handler: getInstanceStates
      InputPayload:
        secondaryTag: '{{SecondaryPatchGroupTag}}'
      Script: |-
        def getInstanceStates(events,context):
          import boto3
```

```
          #Initialize client
          ec2 = boto3.client('ec2')
          tag = events['secondaryTag']
          tagKey, tagValue = list(tag.items())[0]
          instanceQuery = ec2.describe_instances(
          Filters=[
              {
                  "Name": "tag:" + tagKey,
                  "Values": [tagValue]
              }]
          )
          if not instanceQuery['Reservations']:
              noInstancesForTagString = "No instances found for specified tag."
              return({ 'noInstancesFound' : noInstancesForTagString })
          else:
              queryResponse = instanceQuery['Reservations']
              originalInstanceStates = {}
              for results in queryResponse:
                  instanceSet = results['Instances']
                  for instance in instanceSet:
                      instanceId = instance['InstanceId']
                      originalInstanceStates[instanceId] = instance['State']
['Name']
              return originalInstanceStates
    outputs:
      - Name: originalInstanceStates
        Selector: $.Payload
        Type: StringMap
    nextStep: verifySecondaryInstancesRunning
  - name: verifySecondaryInstancesRunning
    action: 'aws:executeScript'
    timeoutSeconds: 600
    onFailure: Abort
    inputs:
      Runtime: python3.7
      Handler: verifyInstancesRunning
      InputPayload:
        targetInstances: '{{getSecondaryInstanceState.originalInstanceStates}}'
      Script: |-
        def verifyInstancesRunning(events,context):
          import boto3

          #Initialize client
          ec2 = boto3.client('ec2')
```

```
          instanceDict = events['targetInstances']
          for instance in instanceDict:
            if instanceDict[instance] == 'stopped':
                print("The target instance " + instance + " is stopped. The
instance will now be started.")
                ec2.start_instances(
                    InstanceIds=[instance]
                    )
            elif instanceDict[instance] == 'stopping':
                print("The target instance " + instance + " is stopping. Polling
for instance to reach stopped state.")
                while instanceDict[instance] != 'stopped':
                    poll = ec2.get_waiter('instance_stopped')
                    poll.wait(
                        InstanceIds=[instance]
                    )
                ec2.start_instances(
                    InstanceIds=[instance]
                )
            else:
              pass
    nextStep: waitForSecondaryRunningInstances
  - name: waitForSecondaryRunningInstances
    action: 'aws:executeScript'
    timeoutSeconds: 300
    onFailure: Abort
    inputs:
      Runtime: python3.7
      Handler: waitForRunningInstances
      InputPayload:
        targetInstances: '{{getSecondaryInstanceState.originalInstanceStates}}'
      Script: |-
        def waitForRunningInstances(events,context):
          import boto3

          #Initialize client
          ec2 = boto3.client('ec2')
          instanceDict = events['targetInstances']
          for instance in instanceDict:
              poll = ec2.get_waiter('instance_running')
              poll.wait(
                  InstanceIds=[instance]
              )
    nextStep: returnSecondaryTagKey
```

```
- name: returnSecondaryTagKey
  action: 'aws:executeScript'
  timeoutSeconds: 120
  onFailure: Abort
  inputs:
    Runtime: python3.7
    Handler: returnTagValues
    InputPayload:
      secondaryTag: '{{SecondaryPatchGroupTag}}'
    Script: |-
      def returnTagValues(events,context):
        tag = events['secondaryTag']
        tagKey = list(tag)[0]
        stringKey = "tag:" + tagKey
        return {'tagKey' : stringKey}
  outputs:
    - Name: Payload
      Selector: $.Payload
      Type: StringMap
    - Name: secondaryPatchGroupKey
      Selector: $.Payload.tagKey
      Type: String
  nextStep: returnSecondaryTagValue
- name: returnSecondaryTagValue
  action: 'aws:executeScript'
  timeoutSeconds: 120
  onFailure: Abort
  inputs:
    Runtime: python3.7
    Handler: returnTagValues
    InputPayload:
      secondaryTag: '{{SecondaryPatchGroupTag}}'
    Script: |-
      def returnTagValues(events,context):
        tag = events['secondaryTag']
        tagKey = list(tag)[0]
        tagValue = tag[tagKey]
        return {'tagValue' : tagValue}
  outputs:
    - Name: Payload
      Selector: $.Payload
      Type: StringMap
    - Name: secondaryPatchGroupValue
      Selector: $.Payload.tagValue
```

```
        Type: String
    nextStep: patchSecondaryInstances
  - name: patchSecondaryInstances
    action: 'aws:runCommand'
    onFailure: Abort
    timeoutSeconds: 7200
    inputs:
      DocumentName: AWS-RunPatchBaseline
      Parameters:
        SnapshotId: '{{SnapshotId}}'
        RebootOption: '{{RebootOption}}'
        Operation: '{{Operation}}'
      Targets:
        - Key: '{{returnSecondaryTagKey.secondaryPatchGroupKey}}'
          Values:
          - '{{returnSecondaryTagValue.secondaryPatchGroupValue}}'
      MaxConcurrency: 10%
      MaxErrors: 10%
    nextStep: returnSecondaryToOriginalState
  - name: returnSecondaryToOriginalState
    action: 'aws:executeScript'
    timeoutSeconds: 600
    onFailure: Abort
    inputs:
      Runtime: python3.7
      Handler: returnToOriginalState
      InputPayload:
        targetInstances: '{{getSecondaryInstanceState.originalInstanceStates}}'
      Script: |-
        def returnToOriginalState(events,context):
          import boto3

          #Initialize client
          ec2 = boto3.client('ec2')
          instanceDict = events['targetInstances']
          for instance in instanceDict:
            if instanceDict[instance] == 'stopped' or instanceDict[instance] ==
'stopping':
                ec2.stop_instances(
                    InstanceIds=[instance]
                    )
            else:
              pass
```

JSON

```
{
         "name":"getSecondaryInstanceState",
         "action":"aws:executeScript",
         "timeoutSeconds":120,
         "onFailure":"Abort",
         "inputs":{
            "Runtime":"python3.7",
            "Handler":"getInstanceStates",
            "InputPayload":{
               "secondaryTag":"{{SecondaryPatchGroupTag}}"
            },
            "Script":"..."
         },
         "outputs":[
            {
               "Name":"originalInstanceStates",
               "Selector":"$.Payload",
               "Type":"StringMap"
            }
         ],
         "nextStep":"verifySecondaryInstancesRunning"
      },
      {
         "name":"verifySecondaryInstancesRunning",
         "action":"aws:executeScript",
         "timeoutSeconds":600,
         "onFailure":"Abort",
         "inputs":{
            "Runtime":"python3.7",
            "Handler":"verifyInstancesRunning",
            "InputPayload":{

 "targetInstances":"{{getSecondaryInstanceState.originalInstanceStates}}"
            },
            "Script":"..."
         },
         "nextStep":"waitForSecondaryRunningInstances"
      },
      {
         "name":"waitForSecondaryRunningInstances",
         "action":"aws:executeScript",
```

```
         "timeoutSeconds":300,
         "onFailure":"Abort",
         "inputs":{
            "Runtime":"python3.7",
            "Handler":"waitForRunningInstances",
            "InputPayload":{

"targetInstances":"{{getSecondaryInstanceState.originalInstanceStates}}"
            },
            "Script":"..."
         },
         "nextStep":"returnSecondaryTagKey"
      },
      {
         "name":"returnSecondaryTagKey",
         "action":"aws:executeScript",
         "timeoutSeconds":120,
         "onFailure":"Abort",
         "inputs":{
            "Runtime":"python3.7",
            "Handler":"returnTagValues",
            "InputPayload":{
               "secondaryTag":"{{SecondaryPatchGroupTag}}"
            },
            "Script":"..."
         },
         "outputs":[
            {
               "Name":"Payload",
               "Selector":"$.Payload",
               "Type":"StringMap"
            },
            {
               "Name":"secondaryPatchGroupKey",
               "Selector":"$.Payload.tagKey",
               "Type":"String"
            }
         ],
         "nextStep":"returnSecondaryTagValue"
      },
      {
         "name":"returnSecondaryTagValue",
         "action":"aws:executeScript",
         "timeoutSeconds":120,
```

```
         "onFailure":"Abort",
         "inputs":{
            "Runtime":"python3.7",
            "Handler":"returnTagValues",
            "InputPayload":{
               "secondaryTag":"{{SecondaryPatchGroupTag}}"
            },
            "Script":"..."
         },
         "outputs":[
            {
               "Name":"Payload",
               "Selector":"$.Payload",
               "Type":"StringMap"
            },
            {
               "Name":"secondaryPatchGroupValue",
               "Selector":"$.Payload.tagValue",
               "Type":"String"
            }
         ],
         "nextStep":"patchSecondaryInstances"
      },
      {
         "name":"patchSecondaryInstances",
         "action":"aws:runCommand",
         "onFailure":"Abort",
         "timeoutSeconds":7200,
         "inputs":{
            "DocumentName":"AWS-RunPatchBaseline",
            "Parameters":{
               "SnapshotId":"{{SnapshotId}}",
               "RebootOption":"{{RebootOption}}",
               "Operation":"{{Operation}}"
            },
            "Targets":[
               {
                  "Key":"{{returnSecondaryTagKey.secondaryPatchGroupKey}}",
                  "Values":[
                     "{{returnSecondaryTagValue.secondaryPatchGroupValue}}"
                  ]
               }
            ],
            "MaxConcurrency":"10%",
```

```
            "MaxErrors":"10%"
         },
         "nextStep":"returnSecondaryToOriginalState"
      },
      {
         "name":"returnSecondaryToOriginalState",
         "action":"aws:executeScript",
         "timeoutSeconds":600,
         "onFailure":"Abort",
         "inputs":{
            "Runtime":"python3.7",
            "Handler":"returnToOriginalState",
            "InputPayload":{

 "targetInstances":"{{getSecondaryInstanceState.originalInstanceStates}}"
            },
            "Script":"..."
         }
      }
   ]
}
```

8. Emily examine le contenu du runbook chiffré terminé et crée le runbook dans le même Compte AWS et la même Région AWS que les instances cibles. Maintenant, elle est prête à tester son runbook afin de s'assurer que l'automatisation fonctionne correctement avant de l'implémenter dans son environnement de production. Voici en quoi consiste le contenu du runbook chiffré terminé.

   YAML

```
description: An example of an Automation runbook that patches groups of Amazon EC2
 instances in stages.
schemaVersion: '0.3'
assumeRole: '{{AutomationAssumeRole}}'
parameters:
  AutomationAssumeRole:
    type: String
    description: '(Required) The Amazon Resource Name (ARN) of the IAM role that
 allows Automation to perform the actions on your behalf. If no role is specified,
 Systems Manager Automation uses your IAM permissions to operate this runbook.'
  PrimaryPatchGroupTag:
    type: StringMap
```

```
    description: '(Required) The tag for the primary group of instances you want
 to patch. Specify a key-value pair. Example: {"key" : "value"}'
  SecondaryPatchGroupTag:
    type: StringMap
    description: '(Required) The tag for the secondary group of instances you want
 to patch. Specify a key-value pair. Example: {"key" : "value"}'
  SnapshotId:
    type: String
    description: '(Optional) The snapshot ID to use to retrieve a patch baseline
 snapshot.'
    default: ''
  RebootOption:
    type: String
    description: '(Optional) Reboot behavior after a patch Install operation. If
 you choose NoReboot and patches are installed, the instance is marked as non-
compliant until a subsequent reboot and scan.'
    allowedValues:
      - NoReboot
      - RebootIfNeeded
    default: RebootIfNeeded
  Operation:
    type: String
    description: '(Optional) The update or configuration to perform on the
 instance. The system checks if patches specified in the patch baseline are
 installed on the instance. The install operation installs patches missing from
 the baseline.'
    allowedValues:
      - Install
      - Scan
    default: Install
mainSteps:
  - name: getPrimaryInstanceState
    action: 'aws:executeScript'
    timeoutSeconds: 120
    onFailure: Abort
    inputs:
      Runtime: python3.7
      Handler: getInstanceStates
      InputPayload:
        primaryTag: '{{PrimaryPatchGroupTag}}'
      Script: |-
        def getInstanceStates(events,context):
          import boto3
```

```
          #Initialize client
          ec2 = boto3.client('ec2')
          tag = events['primaryTag']
          tagKey, tagValue = list(tag.items())[0]
          instanceQuery = ec2.describe_instances(
          Filters=[
              {
                  "Name": "tag:" + tagKey,
                  "Values": [tagValue]
              }]
          )
          if not instanceQuery['Reservations']:
              noInstancesForTagString = "No instances found for specified tag."
              return({ 'noInstancesFound' : noInstancesForTagString })
          else:
              queryResponse = instanceQuery['Reservations']
              originalInstanceStates = {}
              for results in queryResponse:
                  instanceSet = results['Instances']
                  for instance in instanceSet:
                      instanceId = instance['InstanceId']
                      originalInstanceStates[instanceId] = instance['State']
['Name']
              return originalInstanceStates
    outputs:
      - Name: originalInstanceStates
        Selector: $.Payload
        Type: StringMap
    nextStep: verifyPrimaryInstancesRunning
  - name: verifyPrimaryInstancesRunning
    action: 'aws:executeScript'
    timeoutSeconds: 600
    onFailure: Abort
    inputs:
      Runtime: python3.7
      Handler: verifyInstancesRunning
      InputPayload:
        targetInstances: '{{getPrimaryInstanceState.originalInstanceStates}}'
      Script: |-
        def verifyInstancesRunning(events,context):
          import boto3

          #Initialize client
          ec2 = boto3.client('ec2')
```

```
          instanceDict = events['targetInstances']
          for instance in instanceDict:
            if instanceDict[instance] == 'stopped':
                print("The target instance " + instance + " is stopped. The
instance will now be started.")
                ec2.start_instances(
                    InstanceIds=[instance]
                    )
            elif instanceDict[instance] == 'stopping':
                print("The target instance " + instance + " is stopping. Polling
for instance to reach stopped state.")
                while instanceDict[instance] != 'stopped':
                    poll = ec2.get_waiter('instance_stopped')
                    poll.wait(
                        InstanceIds=[instance]
                    )
                ec2.start_instances(
                    InstanceIds=[instance]
                )
            else:
              pass
    nextStep: waitForPrimaryRunningInstances
  - name: waitForPrimaryRunningInstances
    action: 'aws:executeScript'
    timeoutSeconds: 300
    onFailure: Abort
    inputs:
      Runtime: python3.7
      Handler: waitForRunningInstances
      InputPayload:
        targetInstances: '{{getPrimaryInstanceState.originalInstanceStates}}'
      Script: |-
        def waitForRunningInstances(events,context):
          import boto3

          #Initialize client
          ec2 = boto3.client('ec2')
          instanceDict = events['targetInstances']
          for instance in instanceDict:
              poll = ec2.get_waiter('instance_running')
              poll.wait(
                  InstanceIds=[instance]
              )
    nextStep: returnPrimaryTagKey
```

```
  - name: returnPrimaryTagKey
    action: 'aws:executeScript'
    timeoutSeconds: 120
    onFailure: Abort
    inputs:
      Runtime: python3.7
      Handler: returnTagValues
      InputPayload:
        primaryTag: '{{PrimaryPatchGroupTag}}'
      Script: |-
        def returnTagValues(events,context):
          tag = events['primaryTag']
          tagKey = list(tag)[0]
          stringKey = "tag:" + tagKey
          return {'tagKey' : stringKey}
    outputs:
      - Name: Payload
        Selector: $.Payload
        Type: StringMap
      - Name: primaryPatchGroupKey
        Selector: $.Payload.tagKey
        Type: String
    nextStep: returnPrimaryTagValue
  - name: returnPrimaryTagValue
    action: 'aws:executeScript'
    timeoutSeconds: 120
    onFailure: Abort
    inputs:
      Runtime: python3.7
      Handler: returnTagValues
      InputPayload:
        primaryTag: '{{PrimaryPatchGroupTag}}'
      Script: |-
        def returnTagValues(events,context):
          tag = events['primaryTag']
          tagKey = list(tag)[0]
          tagValue = tag[tagKey]
          return {'tagValue' : tagValue}
    outputs:
      - Name: Payload
        Selector: $.Payload
        Type: StringMap
      - Name: primaryPatchGroupValue
        Selector: $.Payload.tagValue
```

```
        Type: String
    nextStep: patchPrimaryInstances
  - name: patchPrimaryInstances
    action: 'aws:runCommand'
    onFailure: Abort
    timeoutSeconds: 7200
    inputs:
      DocumentName: AWS-RunPatchBaseline
      Parameters:
        SnapshotId: '{{SnapshotId}}'
        RebootOption: '{{RebootOption}}'
        Operation: '{{Operation}}'
      Targets:
        - Key: '{{returnPrimaryTagKey.primaryPatchGroupKey}}'
          Values:
            - '{{returnPrimaryTagValue.primaryPatchGroupValue}}'
      MaxConcurrency: 10%
      MaxErrors: 10%
    nextStep: returnPrimaryToOriginalState
  - name: returnPrimaryToOriginalState
    action: 'aws:executeScript'
    timeoutSeconds: 600
    onFailure: Abort
    inputs:
      Runtime: python3.7
      Handler: returnToOriginalState
      InputPayload:
        targetInstances: '{{getPrimaryInstanceState.originalInstanceStates}}'
      Script: |-
        def returnToOriginalState(events,context):
          import boto3

          #Initialize client
          ec2 = boto3.client('ec2')
          instanceDict = events['targetInstances']
          for instance in instanceDict:
            if instanceDict[instance] == 'stopped' or instanceDict[instance] ==
'stopping':
                ec2.stop_instances(
                    InstanceIds=[instance]
                    )
            else:
              pass
    nextStep: getSecondaryInstanceState
```

```
  - name: getSecondaryInstanceState
    action: 'aws:executeScript'
    timeoutSeconds: 120
    onFailure: Abort
    inputs:
      Runtime: python3.7
      Handler: getInstanceStates
      InputPayload:
        secondaryTag: '{{SecondaryPatchGroupTag}}'
      Script: |-
        def getInstanceStates(events,context):
          import boto3

          #Initialize client
          ec2 = boto3.client('ec2')
          tag = events['secondaryTag']
          tagKey, tagValue = list(tag.items())[0]
          instanceQuery = ec2.describe_instances(
          Filters=[
              {
                  "Name": "tag:" + tagKey,
                  "Values": [tagValue]
              }]
          )
          if not instanceQuery['Reservations']:
              noInstancesForTagString = "No instances found for specified tag."
              return({ 'noInstancesFound' : noInstancesForTagString })
          else:
              queryResponse = instanceQuery['Reservations']
              originalInstanceStates = {}
              for results in queryResponse:
                  instanceSet = results['Instances']
                  for instance in instanceSet:
                      instanceId = instance['InstanceId']
                      originalInstanceStates[instanceId] = instance['State']
['Name']
              return originalInstanceStates
    outputs:
      - Name: originalInstanceStates
        Selector: $.Payload
        Type: StringMap
    nextStep: verifySecondaryInstancesRunning
  - name: verifySecondaryInstancesRunning
    action: 'aws:executeScript'
```

```
    timeoutSeconds: 600
    onFailure: Abort
    inputs:
      Runtime: python3.7
      Handler: verifyInstancesRunning
      InputPayload:
        targetInstances: '{{getSecondaryInstanceState.originalInstanceStates}}'
      Script: |-
        def verifyInstancesRunning(events,context):
          import boto3

          #Initialize client
          ec2 = boto3.client('ec2')
          instanceDict = events['targetInstances']
          for instance in instanceDict:
            if instanceDict[instance] == 'stopped':
                print("The target instance " + instance + " is stopped. The
instance will now be started.")
                ec2.start_instances(
                    InstanceIds=[instance]
                    )
            elif instanceDict[instance] == 'stopping':
                print("The target instance " + instance + " is stopping. Polling
for instance to reach stopped state.")
                while instanceDict[instance] != 'stopped':
                    poll = ec2.get_waiter('instance_stopped')
                    poll.wait(
                        InstanceIds=[instance]
                    )
                ec2.start_instances(
                    InstanceIds=[instance]
                )
           else:
             pass
    nextStep: waitForSecondaryRunningInstances
  - name: waitForSecondaryRunningInstances
    action: 'aws:executeScript'
    timeoutSeconds: 300
    onFailure: Abort
    inputs:
      Runtime: python3.7
      Handler: waitForRunningInstances
      InputPayload:
        targetInstances: '{{getSecondaryInstanceState.originalInstanceStates}}'
```

```
    Script: |-
      def waitForRunningInstances(events,context):
        import boto3

        #Initialize client
        ec2 = boto3.client('ec2')
        instanceDict = events['targetInstances']
        for instance in instanceDict:
            poll = ec2.get_waiter('instance_running')
            poll.wait(
                InstanceIds=[instance]
            )
  nextStep: returnSecondaryTagKey
- name: returnSecondaryTagKey
  action: 'aws:executeScript'
  timeoutSeconds: 120
  onFailure: Abort
  inputs:
    Runtime: python3.7
    Handler: returnTagValues
    InputPayload:
      secondaryTag: '{{SecondaryPatchGroupTag}}'
    Script: |-
      def returnTagValues(events,context):
        tag = events['secondaryTag']
        tagKey = list(tag)[0]
        stringKey = "tag:" + tagKey
        return {'tagKey' : stringKey}
  outputs:
    - Name: Payload
      Selector: $.Payload
      Type: StringMap
    - Name: secondaryPatchGroupKey
      Selector: $.Payload.tagKey
      Type: String
  nextStep: returnSecondaryTagValue
- name: returnSecondaryTagValue
  action: 'aws:executeScript'
  timeoutSeconds: 120
  onFailure: Abort
  inputs:
    Runtime: python3.7
    Handler: returnTagValues
    InputPayload:
```

```
        secondaryTag: '{{SecondaryPatchGroupTag}}'
      Script: |-
        def returnTagValues(events,context):
          tag = events['secondaryTag']
          tagKey = list(tag)[0]
          tagValue = tag[tagKey]
          return {'tagValue' : tagValue}
    outputs:
      - Name: Payload
        Selector: $.Payload
        Type: StringMap
      - Name: secondaryPatchGroupValue
        Selector: $.Payload.tagValue
        Type: String
    nextStep: patchSecondaryInstances
  - name: patchSecondaryInstances
    action: 'aws:runCommand'
    onFailure: Abort
    timeoutSeconds: 7200
    inputs:
      DocumentName: AWS-RunPatchBaseline
      Parameters:
        SnapshotId: '{{SnapshotId}}'
        RebootOption: '{{RebootOption}}'
        Operation: '{{Operation}}'
      Targets:
        - Key: '{{returnSecondaryTagKey.secondaryPatchGroupKey}}'
          Values:
          - '{{returnSecondaryTagValue.secondaryPatchGroupValue}}'
      MaxConcurrency: 10%
      MaxErrors: 10%
    nextStep: returnSecondaryToOriginalState
  - name: returnSecondaryToOriginalState
    action: 'aws:executeScript'
    timeoutSeconds: 600
    onFailure: Abort
    inputs:
      Runtime: python3.7
      Handler: returnToOriginalState
      InputPayload:
        targetInstances: '{{getSecondaryInstanceState.originalInstanceStates}}'
      Script: |-
        def returnToOriginalState(events,context):
          import boto3
```

```
          #Initialize client
          ec2 = boto3.client('ec2')
          instanceDict = events['targetInstances']
          for instance in instanceDict:
            if instanceDict[instance] == 'stopped' or instanceDict[instance] ==
 'stopping':
                ec2.stop_instances(
                    InstanceIds=[instance]
                    )
            else:
              pass
```

JSON

```
{
   "description":"An example of an Automation runbook that patches groups of
 Amazon EC2 instances in stages.",
   "schemaVersion":"0.3",
   "assumeRole":"{{AutomationAssumeRole}}",
   "parameters":{
      "AutomationAssumeRole":{
         "type":"String",
         "description":"(Required) The Amazon Resource Name (ARN) of the IAM role
 that allows Automation to perform the actions on your behalf. If no role is
 specified, Systems Manager Automation uses your IAM permissions to operate this
 runbook."
      },
      "PrimaryPatchGroupTag":{
         "type":"StringMap",
         "description":"(Required) The tag for the primary group of instances you
 want to patch. Specify a key-value pair. Example: {\"key\" : \"value\"}"
      },
      "SecondaryPatchGroupTag":{
         "type":"StringMap",
         "description":"(Required) The tag for the secondary group of instances
 you want to patch. Specify a key-value pair. Example: {\"key\" : \"value\"}"
      },
      "SnapshotId":{
         "type":"String",
         "description":"(Optional) The snapshot ID to use to retrieve a patch
 baseline snapshot.",
         "default":""
```

```
      },
      "RebootOption":{
         "type":"String",
         "description":"(Optional) Reboot behavior after a patch Install
operation. If you choose NoReboot and patches are installed, the instance is
marked as non-compliant until a subsequent reboot and scan.",
         "allowedValues":[
            "NoReboot",
            "RebootIfNeeded"
         ],
         "default":"RebootIfNeeded"
      },
      "Operation":{
         "type":"String",
         "description":"(Optional) The update or configuration to perform on
the instance. The system checks if patches specified in the patch baseline are
installed on the instance. The install operation installs patches missing from
the baseline.",
         "allowedValues":[
            "Install",
            "Scan"
         ],
         "default":"Install"
      }
   },
   "mainSteps":[
      {
         "name":"getPrimaryInstanceState",
         "action":"aws:executeScript",
         "timeoutSeconds":120,
         "onFailure":"Abort",
         "inputs":{
            "Runtime":"python3.7",
            "Handler":"getInstanceStates",
            "InputPayload":{
               "primaryTag":"{{PrimaryPatchGroupTag}}"
            },
            "Script":"..."
         },
         "outputs":[
            {
               "Name":"originalInstanceStates",
               "Selector":"$.Payload",
               "Type":"StringMap"
```

```
            }
         ],
         "nextStep":"verifyPrimaryInstancesRunning"
      },
      {
         "name":"verifyPrimaryInstancesRunning",
         "action":"aws:executeScript",
         "timeoutSeconds":600,
         "onFailure":"Abort",
         "inputs":{
            "Runtime":"python3.7",
            "Handler":"verifyInstancesRunning",
            "InputPayload":{

"targetInstances":"{{getPrimaryInstanceState.originalInstanceStates}}"
            },
            "Script":"..."
         },
         "nextStep":"waitForPrimaryRunningInstances"
      },
      {
         "name":"waitForPrimaryRunningInstances",
         "action":"aws:executeScript",
         "timeoutSeconds":300,
         "onFailure":"Abort",
         "inputs":{
            "Runtime":"python3.7",
            "Handler":"waitForRunningInstances",
            "InputPayload":{

"targetInstances":"{{getPrimaryInstanceState.originalInstanceStates}}"
            },
            "Script":"..."
         },
         "nextStep":"returnPrimaryTagKey"
      },
      {
         "name":"returnPrimaryTagKey",
         "action":"aws:executeScript",
         "timeoutSeconds":120,
         "onFailure":"Abort",
         "inputs":{
            "Runtime":"python3.7",
            "Handler":"returnTagValues",
```

```
            "InputPayload":{
               "primaryTag":"{{PrimaryPatchGroupTag}}"
            },
            "Script":"..."
         },
         "outputs":[
            {
               "Name":"Payload",
               "Selector":"$.Payload",
               "Type":"StringMap"
            },
            {
               "Name":"primaryPatchGroupKey",
               "Selector":"$.Payload.tagKey",
               "Type":"String"
            }
         ],
         "nextStep":"returnPrimaryTagValue"
      },
      {
         "name":"returnPrimaryTagValue",
         "action":"aws:executeScript",
         "timeoutSeconds":120,
         "onFailure":"Abort",
         "inputs":{
            "Runtime":"python3.7",
            "Handler":"returnTagValues",
            "InputPayload":{
               "primaryTag":"{{PrimaryPatchGroupTag}}"
            },
            "Script":"..."
         },
         "outputs":[
            {
               "Name":"Payload",
               "Selector":"$.Payload",
               "Type":"StringMap"
            },
            {
               "Name":"primaryPatchGroupValue",
               "Selector":"$.Payload.tagValue",
               "Type":"String"
            }
         ],
```

```
            "nextStep":"patchPrimaryInstances"
         },
         {
            "name":"patchPrimaryInstances",
            "action":"aws:runCommand",
            "onFailure":"Abort",
            "timeoutSeconds":7200,
            "inputs":{
               "DocumentName":"AWS-RunPatchBaseline",
               "Parameters":{
                  "SnapshotId":"{{SnapshotId}}",
                  "RebootOption":"{{RebootOption}}",
                  "Operation":"{{Operation}}"
               },
               "Targets":[
                  {
                     "Key":"{{returnPrimaryTagKey.primaryPatchGroupKey}}",
                     "Values":[
                        "{{returnPrimaryTagValue.primaryPatchGroupValue}}"
                     ]
                  }
               ],
               "MaxConcurrency":"10%",
               "MaxErrors":"10%"
            },
            "nextStep":"returnPrimaryToOriginalState"
         },
         {
            "name":"returnPrimaryToOriginalState",
            "action":"aws:executeScript",
            "timeoutSeconds":600,
            "onFailure":"Abort",
            "inputs":{
               "Runtime":"python3.7",
               "Handler":"returnToOriginalState",
               "InputPayload":{

   "targetInstances":"{{getPrimaryInstanceState.originalInstanceStates}}"
               },
               "Script":"..."
            },
            "nextStep":"getSecondaryInstanceState"
         },
         {
```

```
         "name":"getSecondaryInstanceState",
         "action":"aws:executeScript",
         "timeoutSeconds":120,
         "onFailure":"Abort",
         "inputs":{
            "Runtime":"python3.7",
            "Handler":"getInstanceStates",
            "InputPayload":{
               "secondaryTag":"{{SecondaryPatchGroupTag}}"
            },
            "Script":"..."
         },
         "outputs":[
            {
               "Name":"originalInstanceStates",
               "Selector":"$.Payload",
               "Type":"StringMap"
            }
         ],
         "nextStep":"verifySecondaryInstancesRunning"
      },
      {
         "name":"verifySecondaryInstancesRunning",
         "action":"aws:executeScript",
         "timeoutSeconds":600,
         "onFailure":"Abort",
         "inputs":{
            "Runtime":"python3.7",
            "Handler":"verifyInstancesRunning",
            "InputPayload":{

"targetInstances":"{{getSecondaryInstanceState.originalInstanceStates}}"
            },
            "Script":"..."
         },
         "nextStep":"waitForSecondaryRunningInstances"
      },
      {
         "name":"waitForSecondaryRunningInstances",
         "action":"aws:executeScript",
         "timeoutSeconds":300,
         "onFailure":"Abort",
         "inputs":{
            "Runtime":"python3.7",
```

```
            "Handler":"waitForRunningInstances",
            "InputPayload":{

"targetInstances":"{{getSecondaryInstanceState.originalInstanceStates}}"
            },
            "Script":"..."
         },
         "nextStep":"returnSecondaryTagKey"
      },
      {
         "name":"returnSecondaryTagKey",
         "action":"aws:executeScript",
         "timeoutSeconds":120,
         "onFailure":"Abort",
         "inputs":{
            "Runtime":"python3.7",
            "Handler":"returnTagValues",
            "InputPayload":{
               "secondaryTag":"{{SecondaryPatchGroupTag}}"
            },
            "Script":"..."
         },
         "outputs":[
            {
               "Name":"Payload",
               "Selector":"$.Payload",
               "Type":"StringMap"
            },
            {
               "Name":"secondaryPatchGroupKey",
               "Selector":"$.Payload.tagKey",
               "Type":"String"
            }
         ],
         "nextStep":"returnSecondaryTagValue"
      },
      {
         "name":"returnSecondaryTagValue",
         "action":"aws:executeScript",
         "timeoutSeconds":120,
         "onFailure":"Abort",
         "inputs":{
            "Runtime":"python3.7",
            "Handler":"returnTagValues",
```

```
            "InputPayload":{
               "secondaryTag":"{{SecondaryPatchGroupTag}}"
            },
            "Script":"..."
         },
         "outputs":[
            {
               "Name":"Payload",
               "Selector":"$.Payload",
               "Type":"StringMap"
            },
            {
               "Name":"secondaryPatchGroupValue",
               "Selector":"$.Payload.tagValue",
               "Type":"String"
            }
         ],
         "nextStep":"patchSecondaryInstances"
      },
      {
         "name":"patchSecondaryInstances",
         "action":"aws:runCommand",
         "onFailure":"Abort",
         "timeoutSeconds":7200,
         "inputs":{
            "DocumentName":"AWS-RunPatchBaseline",
            "Parameters":{
               "SnapshotId":"{{SnapshotId}}",
               "RebootOption":"{{RebootOption}}",
               "Operation":"{{Operation}}"
            },
            "Targets":[
               {
                  "Key":"{{returnSecondaryTagKey.secondaryPatchGroupKey}}",
                  "Values":[
                     "{{returnSecondaryTagValue.secondaryPatchGroupValue}}"
                  ]
               }
            ],
            "MaxConcurrency":"10%",
            "MaxErrors":"10%"
         },
         "nextStep":"returnSecondaryToOriginalState"
      },
```

```
      {
         "name":"returnSecondaryToOriginalState",
         "action":"aws:executeScript",
         "timeoutSeconds":600,
         "onFailure":"Abort",
         "inputs":{
            "Runtime":"python3.7",
            "Handler":"returnToOriginalState",
            "InputPayload":{

 "targetInstances":"{{getSecondaryInstanceState.originalInstanceStates}}"
            },
            "Script":"..."
         }
      }
   ]
}
```

Pour de plus amples informations sur les actions d'automatisation utilisées dans cet exemple, veuillez consulter Référence sur les actions Systems Manager Automation.

Exemples supplémentaires de runbook

L'exemple de runbook suivant montre comment vous pouvez utiliser les actions d'automatisation AWS Systems Manager pour automatiser les tâches courantes de déploiement, de dépannage et de maintenance.

> **Note**
>
> Les exemples de runbooks de cette section sont fournis pour démontrer comment créer des runbooks personnalisés pour répondre à vos besoins opérationnels spécifiques. Ces runbooks 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.

Exemples

- Déployer l'architecture VPC et les contrôleurs de domaine Microsoft Active Directory
- Restaurer un volume racine à partir du dernier instantané
- Créer une AMI et une copie inter-régions

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 AWS. L'automatisation des déploiements d'architecture peut également réduire le risque d'erreurs humaines susceptibles de se produire lors du déploiement manuel de l'architecture. AWS Systems Manager Les actions d'automatisation peuvent vous aider à y parvenir. Automation est une fonctionnalité de AWS Systems Manager.

Dans l'exemple suivant, un runbook AWS Systems Manager exécute ces actions :

- Récupère la dernière Amazon Machine Image (AMI) Windows Server 2016 avec Systems Manager Parameter Store à utiliser lors du lancement des instances EC2 qui seront configurées en tant que contrôleurs de domaine. Parameter Store est une fonctionnalité de AWS Systems Manager.
- Utilise l'opération d'automatisation `aws:executeAwsApi` pour appeler plusieurs opérations 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'SSM Agent sur les instances d'accéder aux points de terminaison Systems Manager requis.
- Il utilise l'action d'automatisation `aws:waitForAwsResourceProperty` pour confirmer que les instances lancées par l'action précédente sont `Online` pour AWS Systems Manager.
- Il utilise l'action d'automatisation `aws:runCommand` pour configurer les instances lancées en tant que contrôleurs de domaine Microsoft Active Directory.

YAML

```
---
description: Custom Automation Deployment Example
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 runbook.
mainSteps:
  - name: getLatestWindowsAmi
```

```
        action: aws:executeAwsApi
        onFailure: Abort
        inputs:
          Service: ssm
          Api: GetParameter
          Name: >-
            /aws/service/ami-windows-latest/Windows_Server-2016-English-Full-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 Example",
     "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 run this runbook.",
         "default": ""
       }
     },
     "mainSteps": [
       {
         "name": "getLatestWindowsAmi",
         "action": "aws:executeAwsApi",
         "onFailure": "Abort",
         "inputs": {
           "Service": "ssm",
           "Api": "GetParameter",
           "Name": "/aws/service/ami-windows-latest/Windows_Server-2016-English-
Full-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 l'issue d'une opération d'application de correctifs, le démarrage des instances peut échouer pour cause de noyau ou de 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 d'automatisation peuvent vous aider à y parvenir. Automation est une fonctionnalité de AWS Systems Manager.

Dans l'exemple suivant, un runbook AWS Systems Manager exécute ces actions :

- Il utilise l'action d'automatisation `aws:executeAwsApi` pour récupérer les détails du volume racine de l'instance.
- Il utilise l'action d'automatisation `aws:executeScript` pour récupérer le dernier instantané du volume racine.
- Il utilise l'action d'automatisation `aws:branch` pour poursuivre l'automatisation si un instantané est trouvé pour le volume racine.

YAML

```
   ---
   description: Custom Automation Troubleshooting 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 use this runbook."
       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: python3.8
        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 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 runbook.",
            "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": "$.Volumes[0].VolumeId",
                 "Type": "String"
              }
           ],
           "nextStep": "getSnapshotsByStartTime"
        },
        {
           "name": "getSnapshotsByStartTime",
           "action": "aws:executeScript",
           "timeoutSeconds": 45,
           "onFailure": "Continue",
           "inputs": {
              "Runtime": "python3.8",
              "Handler": "getSnapshotsByStartTime",
              "InputPayload": {
                 "rootVolumeId": "{{ getRootVolumeId.rootVolumeId }}"
              },
              "Attachment": "getSnapshotsByStartTime.py"
           },
           "outputs": [
              {
                 "Name": "Payload",
                 "Selector": "$.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 AWS 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 d'automatisation peuvent vous aider à y parvenir. Automation est une fonctionnalité de AWS Systems Manager.

Dans l'exemple suivant, un runbook AWS Systems Manager exécute ces actions :

- Il utilise l'action d'automatisation `aws:executeAwsApi` pour créer une AMI.
- Il utilise l'action d'automatisation `aws:waitForAwsResourceProperty` pour confirmer la disponibilité de l'AMI.
- Il utilise l'action d'automatisation `aws:executeScript` pour copier l'AMI dans la région de destination.

YAML

```
   ---
   description: Custom Automation Backup and Recovery 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 use this runbook."
       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: python3.8
        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 Example",
      "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 runbook.",
            "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": "python3.8",
              "Handler": "crossRegionImageCopy",
              "InputPayload": {
                 "newImageId": "{{ createImage.newImageId }}"
              },
              "Attachment": "crossRegionImageCopy.py"
           }
        }
     ],
     "files": {
          "crossRegionImageCopy.py": {
              "checksums": {
                  "sha256": "sampleETagValue"
              }
          }
      }
  }
```

Création de paramètres d'entrée qui alimentent les ressources AWS

L'automatisation, une fonctionnalité de Systems Manager, remplit les AWS ressources AWS Management Console qui correspondent au type de ressource que vous définissez pour un paramètre d'entrée. Les ressources de votre Compte AWS qui correspondent au type de ressources défini sont disponibles sous forme de liste déroulante pour vous permettre de les choisir. Vous pouvez définir des types de paramètres d'entrée pour les instances Amazon Elastic Compute Cloud (Amazon EC2), les buckets Amazon Simple Storage Service (Amazon S3) et les rôles (IAM). AWS Identity and Access Management Les définitions de types prises en charge et les expressions régulières utilisées pour localiser les ressources correspondantes sont les suivantes :

- `AWS::EC2::Instance::Id` - `^m?i-([a-z0-9]{8}|[a-z0-9]{17})$`
- `List<AWS::EC2::Instance::Id>` - `^m?i-([a-z0-9]{8}|[a-z0-9]{17})$`
- `AWS::S3::Bucket::Name` - `^[0-9a-z][a-z0-9\\-\\.]{3,63}$`
- `List<AWS::S3::Bucket::Name>` - `^[0-9a-z][a-z0-9\\-\\.]{3,63}$`
- `AWS::IAM::Role::Arn` - `^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):iam::[0-9]{12}:role/.*$`
- `List<AWS::IAM::Role::Arn>` - `^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):iam::[0-9]{12}:role/.*$`

Voici un exemple de types de paramètres d'entrée définis dans le contenu d'un runbook.

YAML

```
description: Enables encryption on an Amazon S3 bucket
schemaVersion: '0.3'
assumeRole: '{{ AutomationAssumeRole }}'
parameters:
  BucketName:
    type: 'AWS::S3::Bucket::Name'
    description: (Required) The name of the Amazon S3 bucket you want to encrypt.
  SSEAlgorithm:
    type: String
    description: (Optional) The server-side encryption algorithm to use for the
 default encryption.
    default: AES256
  AutomationAssumeRole:
    type: 'AWS::IAM::Role::Arn'
```

```
    description: (Optional) The Amazon Resource Name (ARN) of the role that allows
 Automation to perform the actions on your behalf.
    default: ''
mainSteps:
  - name: enableBucketEncryption
    action: 'aws:executeAwsApi'
    inputs:
      Service: s3
      Api: PutBucketEncryption
      Bucket: '{{BucketName}}'
      ServerSideEncryptionConfiguration:
        Rules:
          - ApplyServerSideEncryptionByDefault:
              SSEAlgorithm: '{{SSEAlgorithm}}'
    isEnd: true
```

JSON

```
{
   "description": "Enables encryption on an Amazon S3 bucket",
   "schemaVersion": "0.3",
   "assumeRole": "{{ AutomationAssumeRole }}",
   "parameters": {
      "BucketName": {
         "type": "AWS::S3::Bucket::Name",
         "description": "(Required) The name of the Amazon S3 bucket you want to
 encrypt."
      },
      "SSEAlgorithm": {
         "type": "String",
         "description": "(Optional) The server-side encryption algorithm to use for
 the default encryption.",
         "default": "AES256"
      },
      "AutomationAssumeRole": {
         "type": "AWS::IAM::Role::Arn",
         "description": "(Optional) The Amazon Resource Name (ARN) of the role that
 allows Automation to perform the actions on your behalf.",
         "default": ""
      }
   },
   "mainSteps": [
      {
```

```
         "name": "enableBucketEncryption",
         "action": "aws:executeAwsApi",
         "inputs": {
            "Service": "s3",
            "Api": "PutBucketEncryption",
            "Bucket": "{{BucketName}}",
            "ServerSideEncryptionConfiguration": {
               "Rules": [
                  {
                     "ApplyServerSideEncryptionByDefault": {
                        "SSEAlgorithm": "{{SSEAlgorithm}}"
                     }
                  }
               ]
            }
         },
         "isEnd": true
      }
   ]
}
```

Créer des runbooks à l'aide de Document Builder

Si les runbooks AWS Systems Manager publics ne prennent pas en charge toutes les actions que vous souhaitez effectuer sur vos AWS ressources, vous pouvez créer vos propres runbooks. Pour créer un runbook personnalisé, vous pouvez créer manuellement un fichier local au format YAML ou JSON contenant les actions d'automatisation appropriées. Vous pouvez également utiliser Document Builder dans la console Systems Manager Automation pour créer un runbook personnalisé.

Grâce à Document Builder, vous pouvez ajouter des actions d'automatisation à votre runbook personnalisé et fournir les paramètres requis, sans utiliser la syntaxe JSON ou YAML. Après avoir ajouté des étapes et créé le runbook, le système convertit les actions que vous avez ajoutées au format YAML que Systems Manager peut utiliser pour exécuter l'automatisation.

Les runbooks prennent en charge l'utilisation de Markdown, un langage de balisage, qui vous permet d'ajouter des descriptions de style wiki aux runbooks et des étapes individuelles au sein du runbook. Pour plus d'informations sur l'utilisation de Markdown, consultez Utilisation de Markdown dans AWS.

Créer un runbook à l'aide de Document Builder

Avant de commencer

Nous vous recommandons de prendre connaissance des différentes actions que vous pouvez effectuer dans un runbook. Pour plus d'informations, consultez Référence sur les actions Systems Manager Automation.

Pour créer un runbook à l'aide de Document Builder

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.
3. Sélectionnez Create automation (Créer une automatisation).
4. Pour Name (Nom), saisissez un nom descriptif pour le runbook.
5. Pour Document description (Description du document), indiquez la description du style de balisage pour le runbook. Vous pouvez fournir des instructions sur l'utilisation du runbook, des étapes numérotées ou tout autre type d'information pour décrire le runbook. 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 exécute l'automatisation.

   > **Important**
   >
   > Pour les runbooks qui ne sont pas la propriété d'Amazon et qui utilisent l'action `aws:executeScript`, un rôle doit être spécifié. Pour plus d'informations, veuillez consulter Autorisations pour l'utilisation de runbooks.

7. (Facultatif) Pour Outputs (Sorties), saisissez une sortie pour l'automatisation de ce runbook à mettre à disposition d'autres processus.

   Par exemple, si votre runbook crée une nouvelle AMI, vous pouvez spécifier [». CreateImage ImageId«], puis utilisez cette sortie pour créer de nouvelles instances dans le cadre d'une automatisation ultérieure.

8. (Facultatif) Développez la section Input Parameters (Paramètres d'entrée) et procédez comme suit.

   1. Dans Parameter name (Nom de paramètre), saisissez un nom descriptif pour le paramètre de runbook que vous créez.
   2. Pour Type, sélectionnez un type pour le paramètre, par exemple, `String` ou `MapList`.
   3. Pour Required (Obligatoire), effectuez l'une des opérations suivantes :
      - Sélectionnez Yes (Oui) si une valeur pour ce paramètre de runbook doit être fournie lors de l'exécution.
      - Sélectionnez No (Non) si le paramètre n'est pas obligatoire et (facultatif) saisissez une valeur de paramètre par défaut dans Defaults value (Valeur par défaut).
   4. Dans Description, saisissez une description pour le paramètre de runbook.

   > **Note**
   >
   > Pour ajouter d'autres paramètres de runbook, sélectionnez Add a parameter (Ajouter un paramètre). Pour supprimer un paramètre de runbook, cliquez sur le bouton X (Supprimer).

9. (Facultatif) Développez la section Target type (Type de cible) et sélectionnez un type de cible pour définir les types de ressources sur lesquelles l'automatisation peut s'exécuter. Par exemple, pour exécuter un runbook sur des instance EC2, sélectionnez `/AWS::EC2::Instance`.

   > **Note**
   >
   > Si vous spécifiez la valeur « `/` », le runbook 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 Guide de l'utilisateur AWS CloudFormation .

10. (Facultatif) Développez la section Document tags (Balises de document) et saisissez une ou plusieurs paires clé-valeur de balise à appliquer au runbook. Les balises facilitent l'identification, l'organisation et la recherche de ressources. Pour plus d'informations, consultez Balisage des documents Systems Manager.
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 de l'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.
- 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 correspondant au type d'action sélectionné. Par exemple : `aws:executeStateMachine` – exécuter une machine d'état AWS Step Functions.
- (Facultatif) Pour Additional inputs (entrées supplémentaires), indiquez toutes les valeurs d'entrée supplémentaires nécessaires à votre runbook. 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, sélectionnez 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 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 d'automatisation. 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 Propriétés partagées par toutes les actions.

> **Note**
>
> Pour ajouter d'autres étapes, sélectionnez Add step (Ajouter étape) et répétez la procédure de création d'une étape. Pour supprimer une étape, sélectionnez Remove step (Supprimer étape).

12. Sélectionnez Create automation (Créer une automatisation) pour enregistrer le runbook.

Créer un runbook qui exécute des scripts

La procédure suivante explique comment utiliser Document Builder dans la console AWS Systems Manager Automation pour créer un runbook personnalisé qui exécute un script.

La première étape du runbook que vous créez exécute un script pour lancer une instance Amazon Elastic Compute Cloud (Amazon 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'automatisation.

Avant de commencer

Assurez-vous que vous avez suivi les étapes ci-dessous :

- 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, veuillez consulter Vérification de l'accès utilisateur aux runbooks.
- 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 automatisations.

  Pour plus d'informations sur le rôle de service IAM requis pour exécuter `aws:executeScript`, consultez Autorisations pour l'utilisation de runbooks.

- Vérifiez que vous avez l'autorisation de lancer des instances EC2.

  Pour plus d'informations, consultez IAM et Amazon EC2 dans le guide de l'utilisateur Amazon EC2.

Créer un runbook personnalisé qui exécute des scripts à l'aide de Document Builder

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.
3. Sélectionnez Create automation (Créer une automatisation).
4. Dans Name (Nom), saisissez ce nom descriptif pour le runbook : **`LaunchInstanceAndCheckStatus`**.
5. (Facultatif) Pour Document description (Description du document), remplacez le texte par défaut par une description pour ce runbook, à l'aide de Markdown. Voici un exemple.

```
##Title: LaunchInstanceAndCheckState
    -----
    **Purpose**: This runbook first launches an EC2 instance using the AMI
 ID provided in the parameter ```imageId```. The second step of this runbook
 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 de responsable), saisissez l'ARN du rôle de service IAM pour Automation (rôle de responsable) pour l'exécution de l'automatisation, au format **`arn:aws:iam::111122223333:role/AutomationServiceRole`**. Remplacez votre Compte AWS identifiant par 111122223333.

   Le rôle que vous spécifiez est utilisé pour fournir les autorisations nécessaires pour démarrer l'automatisation.

> **Important**
>
> Pour les runbooks qui ne sont pas la propriété d'Amazon et qui utilisent l'action `aws:executeScript`, un rôle doit être spécifié. Pour plus d'informations, veuillez consulter Autorisations pour l'utilisation de runbooks.

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, sélectionnez **`String`**.
   3. Pour Required (Obligatoire), sélectionnez 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 Amazon EC2 en utilisant le dernier identifiant Amazon Linux 1 Amazon Machine Image (AMI). 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. Sélectionnez 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, sélectionnez **`String`**.
   3. Pour Required (Obligatoire), sélectionnez No.
   4. Pour Default value (Valeur par défaut), entrez **`LaunchedBySsmAutomation`**. La paire clé-valeur de balise `Name:LaunchedBySsmAutomation` est ajoutée 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. Sélectionnez 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, sélectionnez **`String`**.
    3. Pour Required (Obligatoire), sélectionnez 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 sélectionnez **`"/"`**.
11. (Facultatif) Développez les Document tags (Balises de document) pour appliquer des balises de ressources à votre runbook. 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 un nom descriptif pour la première étape de l'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 Runtime, sélectionnez 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**
>
> Cela 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. Dans Nom de l'entrée, choisissez InputPayload. Pour Input value (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 Name (Nom), saisissez **`payload`**.
    - Pour Selector (Sélecteur), entrez **`$.Payload`**.
    - Dans le champ Type, sélectionnez `StringMap`.

14. Sélectionnez Add step (Ajouter une étape) pour ajouter une deuxième étape au runbook. 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 un nom descriptif pour la deuxième étape de l'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, sélectionnez 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**
>
> Cela 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. Dans Nom de l'entrée, choisissez InputPayload. Pour Input value (Valeur d'entrée), entrez ce qui suit.

```
{{ LaunchEc2Instance.payload }}
```

16. Sélectionnez Create automation (Créer une automatisation) pour enregistrer le runbook.

Utilisation de scripts dans des runbooks

Les runbooks Automation prennent en charge l'exécution de scripts dans le cadre de l'automatisation. Automation est une fonctionnalité de AWS Systems Manager. En utilisant les runbooks, vous pouvez exécuter des scripts directement dans AWS sans créer un environnement de calcul distinct pour exécuter vos scripts. Comme les runbooks 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. Vous pouvez envoyer la sortie depuis des actions `aws:executeScript` de vos runbooks vers Amazon CloudWatch Logs. Pour de plus amples informations, veuillez consulter Journalisation de la sortie d'actions Automation avec CloudWatch Logs.

Autorisations pour l'utilisation de runbooks

Pour utiliser un runbook, 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 runbooks 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 runbook ou transmis en tant que paramètre.
- Si aucun rôle de service IAM n'est spécifié, il utilise les autorisations de l'utilisateur qui a lancé l'automatisation.

Toutefois, lorsqu'une étape d'un runbook comprend l'action `aws:executeScript`, un rôle de service IAM (rôle de responsable) est toujours requis si le script Python ou PowerShell spécifié pour l'action appelle des opérations d'API AWS. 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 runbook ou transmis en tant que paramètre.
- Si aucun rôle n'est trouvé, Automation tente d'exécuter le script Python ou PowerShell 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.

Ajout de scripts à des runbooks

Vous pouvez ajouter des scripts à vos runbooks en incluant le script en ligne comme partie d'une étape du runbook. Vous pouvez également joindre des scripts au runbook 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 runbook.

Contraintes de script applicables à des runbooks

Les runbooks imposent une limite de cinq fichiers en pièce jointe. Les scripts peuvent être un script Python (.py), un script PowerShell Core (.ps1), ou ils peuvent être joints sous forme de contenu dans un fichier .zip.

Utilisation d'instructions conditionnelles dans les runbooks

Par défaut, les étapes que vous définissez dans la section `mainSteps` d'un runbook 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 de l'automatisation échoue (par défaut). Vous pouvez utiliser les options d'action d'automatisation `aws:branch` et de runbook décrites dans cette section pour créer des automatisations qui effectuent des ramifications conditionnelles. Autrement dit, vous pouvez créer des automatisations 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 automatisations dynamiques :

- **`aws:branch`**: cette action d'automatisation vous permet de créer une automatisation dynamique qui évalue plusieurs options en une seule étape, puis passe à une autre étape dans le runbook Automation en fonction des résultats de cette évaluation.
- **`nextStep`**: cette option spécifie l'étape d'une automatisation à traiter immédiatement après la fin d'une étape.
- **`isEnd`**: cette option arrête une exécution d'automatisation 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 de l'automatisation. Si une étape portant cette désignation échoue, l'automatisation signale l'état final de l'automatisation comme `Failed`. La valeur par défaut de cette option est `true`.

- **`onFailure`**: cette option indique si l'automatisation doit être arrêtée, poursuivie ou changer d'étape en cas d'échec. La valeur par défaut de cette option est abort.

La section suivante décrit l'action de l'automatisation `aws:branch`. Pour de plus amples informations sur les options de `nextStep`, `isEnd`, `isCritical` et `onFailure`, veuillez consulter Exemple de runbooks `aws:branch`.

Utilisation de l'action **`aws:branch`**

L'action `aws:branch` propose les options les plus dynamiques des ramifications conditionnelles pour les automatisations. Comme noté précédemment, cette action permet à vos automatisations 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 l'automatisation doit évaluer. L'automatisation peut évaluer ces `Choices` en fonction de la valeur d'un paramètre que vous avez spécifiée dans la section `Parameters` du runbook. L'automatisation peut également évaluer ces `Choices` en fonction de la sortie d'une étape précédente.

L'automatisation évalue chaque choix à l'aide d'une expression booléenne. Si l'évaluation détermine que le premier choix est `true`, l'automatisation accède à l'étape désignée pour ce choix. Si l'évaluation détermine que le premier choix est `false`, l'automatisation évalue le choix suivant. Si l'étape inclut au moins `Choices`, l'automatisation évalue chaque choix par ordre séquentiel jusqu'à

ce qu'il en identifie un possédant la valeur `true`. L'automatisation accède ensuite à l'étape désignée correspondant au choix défini sur `true`.

Si aucun des `Choices` n'est `true`, l'automatisation vérifie si l'étape contient une valeur `Default`. La valeur `Default` définit une étape à laquelle l'automatisation doit passer si aucun des choix n'est défini sur `true`. Si aucune valeur `Default` n'est spécifiée pour l'étape, l'automatisation traite l'étape suivante du runbook.

Voici une `aws:branch` étape en YAML nommée SfromParameterChooSEO. Cette étape inclut deux `Choices` : (`NextStep: runWindowsCommand`) et (`NextStep: runLinuxCommand`). L'automatisation 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 runbook.

```
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 en YAML nommée SfromOutputChooSEO. Cette étape inclut deux `Choices` : (`NextStep: runPowerShellCommand`) et (`NextStep: runShellCommand`). L'automatisation 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 runbook. Cet exemple inclut également une option appelée `Default`. Si l'automatisation évalue les deux `Choices`, et si aucun choix n'est `true`, l'automatisation accède à une étape nommée `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 runbook

Lorsque vous créez une étape `aws:branch` dans un runbook, vous définissez les `Choices` que l'automatisation doit évaluer pour déterminer l'étape suivante à laquelle elle doit passer. Comme indiqué précédemment, l'évaluation des `Choices` repose sur une expression booléenne. Chaque choix doit définir les options suivantes :

- NextStep: étape suivante du runbook à traiter si le choix indiqué est`true`.
- Variable : Spécifiez soit le nom d'un paramètre défini dans la `Parameters` section du runbook, soit une variable définie dans la `Variables` section, soit un objet de sortie issu d'une étape précédente.

  Spécifiez les valeurs des variables à l'aide du formulaire suivant.

  `Variable: "{{`*`variable name`*`}}"`

  Spécifiez les valeurs des paramètres à l'aide du formulaire suivant.

  `Variable: "{{`*`parameter name`*`}}"`

  Spécifiez les variables d'objet de sortie sous la forme suivante.

  `Variable: "{{`*`previousStepName.outputName`*`}}"`

  > **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.

- Opération : critères utilisés pour évaluer le choix, par exemple `StringEquals: Linux`. 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 runbook, le système valide chaque opération dans le runbook. Si une opération n'est pas prise en charge, le système renvoie une erreur lorsque vous tentez de créer le runbook.

- Default : spécifiez une étape de rechange à laquelle l'automatisation doit passer si aucun des `Choices` n'est défini sur `true`.

> **Note**
>
> Si vous ne voulez pas spécifier de valeur `Default`, vous pouvez spécifier l'option `isEnd`. Si aucun des `Choices` n'est `true` et qu'aucune valeur `Default` n'est spécifiée, l'automatisation s'arrête à la fin de l'étape.

Utilisez les modèles suivants pour vous aider à construire l'étape `aws:branch` dans votre runbook : Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations.

YAML

```
mainSteps:
- name: step name
  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 for this choice is 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 for this choice is
 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.outputName`*`}}"`

Par exemple, dans l'exemple suivant, la première étape est nommée `GetInstance`. Ensuite, sous `outputs`, figure un champ appelé `platform`. Dans la deuxième étape (`ChooseOSforCommands`), l'auteur souhaite référencer la sortie du champ de la plateforme en tant que variable. Pour créer la variable, il suffit de combiner le nom de l'étape (GetInstance) et le nom du champ de sortie (plateforme) à 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 qui montre comment *"Variable": "{{ describeInstance.Platform }}"* est créée à partir de l'étape précédente et du résultat.

```
- 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 runbooks **`aws:branch`**

Voici quelques exemples de runbooks utilisant `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 (`GetInstance`), le créateur du runbook utilise l'action `aws:executeAwsApi` pour appeler l'opération d'API `DescribeInstanceInformation` de `ssm`. 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`). L'automatisation évalue le système d'exploitation de l'instance en utilisant la sortie de l'étape précédente (`Variable: "{{GetInstance.platform}}"`). L'automatisation 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 runbook dans la section `parameters`. Un paramètre est nommé `OperatingSystemName`. Dans la première étape (`ChooseOS`), l'auteur utilise l'action `aws:branch` 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 runbook, il spécifie une valeur au moment de l'exécution pour `OperatingSystemName`. L'automatisation utilise le paramètre d'exécution lors de l'évaluation `Choices`. L'automatisation 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 d'automatisations à ramifications complexes avec des opérateurs

Vous pouvez créer des automatisations 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, l'automatisation accède à l'étape `runPowerShellCommand`. Si une ou plusieurs des variables sont `false`, l'automatisation 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, l'automatisation accède à l'étape `RunPowerShellCommand`. Si les deux variables sont false, l'automatisation 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 un paramètre de chaîne est `Not Linux`. Si l'évaluation détermine que la variable ne correspond pas à Linux, l'automatisation accède à l'étape `sleep2`. Si l'évaluation du premier choix détermine que la variable est Linux, l'automatisation 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 conditionnelles

Cette section inclut différents exemples d'utilisation des options dynamiques dans un runbook. Chaque exemple présenté dans cette section étend le runbook suivant. Ce runbook comporte deux actions. La première action se nomme `InstallMsiPackage`. Il utilise l'action `aws:runCommand`

pour installer une application sur une instance Windows Server. La deuxième action se nomme `TestInstall`. Elle utilise l'action `aws:invokeLambdaFunction` pour effectuer un test de l'application installée si l'application a été installée correctement. La première étape spécifie `onFailure: Abort`. Ainsi, si l'application ne s'est pas installée correctement, l'automatisation s'arrête avant la deuxième étape.

Exemple 1 : runbook comportant 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:
    - "{{instanceIds}}"
    DocumentName: AWS-RunPowerShellScript
    Parameters:
      commands:
      - msiexec /i {{packageName}}
- name: TestInstall
  action: aws:invokeLambdaFunction
  maxAttempts: 1
  timeoutSeconds: 500
  inputs:
    FunctionName: TestLambdaFunction
...
```

Création d'une automatisation dynamique qui passe à différentes étapes à l'aide de l'option **onFailure**

L'exemple suivant utilise les options onFailure: step:*step name*, nextStep et isEnd pour créer une automatisation dynamique. Dans cet exemple, si l'InstallMsiPackageaction échoue, l'automatisation passe à une action appelée PostFailure(onFailure: step:PostFailure) pour exécuter une AWS Lambda fonction afin d'effectuer une action en cas d'échec de l'installation. Si l'installation réussit, l'automatisation passe à l' TestInstall action ()nextStep: TestInstall. Les étapes TestInstall et PostFailure utilisent l'option isEnd (isEnd: true) pour que l'automatisation se termine 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, l'automatisation s'arrête après l'exécution de l'action effectuée à la dernière étape.

Exemple 2 : automatisation dynamique qui passe à différentes étapes

```
mainSteps
- name: InstallMsiPackage
  action: aws:runCommand
  onFailure: step:PostFailure
  maxAttempts: 2
  inputs:
    InstanceIds:
    - "{{instanceIds}}"
    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 runbook, le système vérifie que le runbook 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'une automatisation dynamique qui définit des étapes critiques

Vous pouvez spécifier qu'une étape est critique pour la réussite globale de l'automatisation. Si une étape critique échoue, Automation signale l'état `Failed` pour l'automatisation, même si une ou plusieurs étapes se sont déroulées correctement. Dans l'exemple suivant, l'utilisateur identifie l'VerifyDependenciesétape en cas d'échec (`onFailure: step:VerifyDependencies`). InstallMsiPackage 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 de l'automatisation

```
---
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
...
```

Utilisation des sorties d'action comme entrées

Plusieurs actions d'automatisation renvoient des résultats prédéfinis. Vous pouvez transmettre ces sorties en tant qu'entrées aux étapes ultérieures de votre runbook en utilisant le format `{{stepName.outputName}}`. Vous pouvez définir des sorties pour différentes actions d'automatisation dans vos runbooks. Cela vous permet d'exécuter des scripts ou d'appeler des opérations d'API pour Services AWS une autre fois afin de pouvoir réutiliser les valeurs comme entrées lors d'actions ultérieures. Les types de paramètres des runbooks sont statiques. Cela signifie que le type de paramètre ne peut pas être modifié après avoir été défini. Pour définir une sortie d'étape, fournir les champs suivants :

- Nom : (Obligatoire) le nom de sortie utilisé pour faire référence à la valeur de la sortie dans les étapes ultérieures.
- Sélecteur : (Obligatoire) expression JSONPath utilisée pour déterminer la valeur de sortie.
- Type : (Facultatif) type de données de la valeur renvoyée par le champ de sélection. Les valeurs de type valides sont `String`, `Integer`, `Boolean`, `StringList`, `StringMap`, `MapList`. La valeur par défaut est `String`.

Si la valeur d'une sortie ne correspond pas au type de données que vous avez spécifié, l'Automatisation essaie de convertir le type de données. Par exemple, si la valeur renvoyée est `Integer`, mais que le `Type` spécifié est `String`, la valeur de sortie finale est une valeur `String`. Les types de conversions suivants sont pris en charge :

- Les valeurs `String` peuvent être converties en `StringList`, `Integer` et `Boolean`.
- Les valeurs `Integer` peuvent être converties en `String` et `StringList`.
- Les valeurs `Boolean` peuvent être converties en `String` et `StringList`.
- Les valeurs `StringList`, `IntegerList` ou `BooleanList` contenant un élément peuvent être converties en `String`, `Integer` ou `Boolean`.

Lorsque vous utilisez des paramètres ou des sorties avec des actions d'automatisation, le type de donnée ne peut pas être modifié dynamiquement dans l'entrée d'une action.

Voici un exemple de runbook qui montre comment définir les sorties d'action et référencer la valeur en tant qu'entrée pour une action ultérieure. Les runbooks réalisent les opérations suivantes :

- Utilise l'`aws:executeAwsApi`action pour appeler l'opération d' DescribeImages API Amazon EC2 afin d'obtenir le nom d'un Windows Server 2016 spécifique. AMI Il génère l'ID d'image en tant que `ImageId`.
- Utilise l'`aws:executeAwsApi`action pour appeler l'opération d' RunInstances API Amazon EC2 afin de lancer une instance qui utilise celle `ImageId` de l'étape précédente. Il génère l'ID d'instance en tant que `InstanceId`.
- Utilise l' `aws:waitForAwsResourceProperty`action pour interroger le fonctionnement de l' DescribeInstanceStatus API Amazon EC2 afin d'attendre que l'instance atteigne son état. `running` 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.
- Utilise l'action `aws:assertAwsResourceProperty` pour appeler l'opération d'API `DescribeInstanceStatus` d'Amazon EC2 afin d'affirmer que l'instance a le statut `running`. L'étape échoue si l'état de l'instance n'est pas `running`.

```
---
description: Sample runbook using AWS API operations
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-2022-English-Full-Base*"
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
  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"
...
```

Chacune des actions d'automatisation décrites précédemment vous permet d'appeler une opération d'API spécifique en définissant l'espace de noms de service, l'opération d'API, les paramètres d'entrée et les paramètres de sortie. Les entrées sont définies par l'opération d'API que vous

sélectionnez. Vous pouvez consulter les opérations 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. Sélectionnez une méthode dans la section Client pour le service que vous voulez appeler. Par exemple, toutes les opérations d'API (méthodes) pour Amazon Relational Database Service (Amazon RDS) sont répertoriées à la page suivante : Méthodes pour Amazon RDS.

Vous pouvez consulter le schéma de chaque action d'automatisation dans les emplacements suivants :

- `aws:assertAwsResourceProperty` - Affirmer un statut de ressource AWS ou un statut d'événement
- `aws:executeAwsApi`— Appelez et exécutez des opérations AWS d'API
- `aws:waitForAwsResourceProperty` - Attendre sur une propriété de ressource AWS

Les schémas incluent les descriptions des champs obligatoires pour l'utilisation de chaque action.

Utilisation du sélecteur ou PropertySelector des champs

Chaque action Automation exige que vous spécifiiez une sortie `Selector` (pour `aws:executeAwsApi`) ou `PropertySelector` (pour `aws:assertAwsResourceProperty` et `aws:waitForAwsResourceProperty`). Ces champs sont utilisés pour traiter la réponse JSON d'une opération d' AWS API. Ces champs utilisent la syntaxe JSONPath.

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"
...
```

Dans l'étape `getImageId` `aws:executeAwsApi`, l'automatisation appelle l'opération d'API `DescribeImages` et reçoit une réponse d'`ec2`. L'automatisation applique ensuite `Selector - "$.Images[0].ImageId"` à la réponse de l'API et attribue la valeur sélectionnée pour à la variable `ImageId` de sortie. D'autres étapes dans la même 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
...
```

Dans l'étape `waitUntilInstanceStateRunning` `aws:waitForAwsResourceProperty`, l'automatisation appelle l'opération d'API `DescribeInstanceStatus` et reçoit une réponse d'`ec2`. L'automatisation applique ensuite `PropertySelector - "$.InstanceStatuses[0].InstanceState.Name"` à la réponse et vérifie si la valeur renvoyée correspond à une valeur spécifiée dans la liste `DesiredValues` (dans ce cas, `running`). L'étape répète le processus jusqu'à ce que la réponse renvoie l'état d'instance `running`.

Utilisation de JSONPath dans des runbooks

Une expression JSONPath est une chaîne commençant par « $. » qui est utilisée pour sélectionner un ou plusieurs composants d'un élément JSON. La liste suivante inclut des informations sur les opérateurs JSONPath qui sont pris en charge par Systems Manager Automation :

- Enfant à notation point (.) : à utiliser avec un objet JSON. Cet opérateur sélectionne la valeur d'une clé spécifique.
- Analyse approfondie (..) : à utiliser avec un élément JSON. Cet opérateur numérise l'élément JSON niveau par niveau et sélectionne une liste de valeurs avec la clé spécifique. Le type de retour de cet opérateur est toujours un tableau JSON. Dans le contexte d'un type de sortie d'action d'automatisation, l'opérateur peut être l'un StringList ou l'autre MapList.
- Index de tableau ([]) : à utiliser avec un tableau JSON. Cet opérateur obtient la valeur d'un index spécifique.
- Filtre ([?(***expression***)]) : à utiliser avec un tableau JSON. Cet opérateur filtre les valeurs des tableaux JSON qui correspondent aux critères définis dans l'expression du filtre. Les expressions de filtre ne peuvent utiliser que les opérateurs suivants : **==**, **! =**, **>**, **<**, **>=** ou **<=**. La combinaison de plusieurs expressions de filtre avec AND (&&) ou OR (||) n'est pas prise en charge. Le type de retour de cet opérateur est toujours un tableau JSON.

Pour mieux comprendre les opérateurs JSONPath, examinez la réponse JSON suivante à partir de l'opération d'API EC2 `DescribeInstances`. À la suite de cette réponse figurent plusieurs exemples qui illustrent différents résultats en appliquant différentes expressions JSONPath à la réponse à partir de l'opération d'API `DescribeInstances`.

```
{
    "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": []
        }
    ]
}
```

Exemple JSONPath 1 : obtention d'une chaîne spécifique à partir d'une réponse JSON

```
JSONPath:
$.Reservations[0].Instances[0].ImageId

Returns:
"ami-12345678"

Type: String
```

Exemple JSONPath 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
```

Exemple JSONPath 3 : obtention d'un entier spécifique à partir d'une réponse JSON

```
JSONPath:
$.Reservations[0].Instances[0].State.Code

Returns:
16

Type: Integer
```

Exemple 4 de JSONPath : analysez en profondeur une réponse JSON, puis obtenez toutes les valeurs de VolumeId StringList

```
JSONPath:
$.Reservations..BlockDeviceMappings..VolumeId

Returns:
[
   "vol-000000000000",
   "vol-111111111111"
]

Type: StringList
```

Exemple JSONPath 5 : Obtenir un BlockDeviceMappings objet spécifique en tant que StringMap

```
JSONPath:
$.Reservations[0].Instances[0].BlockDeviceMappings[0]

Returns:
{
```

```
   "Ebs" : {
      "DeleteOnTermination" : true,
      "Status" : "attached",
      "VolumeId" : "vol-000000000000"
   },
   "DeviceName" : "/dev/xvda"
}

Type: StringMap
```

Exemple 6 de JSONPath : analyse en profondeur une réponse JSON, puis récupère tous les objets State sous forme de MapList

```
JSONPath:
$.Reservations..Instances..State

Returns:
[
   {
      "Code" : 16,
      "Name" : "running"
   },
   {
      "Code" : 80,
      "Name" : "stopped"
   }
]

Type: MapList
```

JSONPath exemple 7 : filtre pour les instances dans l'état **`running`**

```
JSONPath:
$.Reservations..Instances[?(@.State.Name == 'running')]

Returns:
[
  {
    "ImageId": "ami-12345678",
    "BlockDeviceMappings": [
      {
        "Ebs": {
          "DeleteOnTermination": true,
```

```
          "Status": "attached",
          "VolumeId": "vol-000000000000"
        },
        "DeviceName": "/dev/xvda"
      }
    ],
    "State": {
      "Code": 16,
      "Name": "running"
    }
  }
]

Type: MapList
```

JSONPath exemple 8 : renvoie le **`ImageId`** des instances qui ne sont pas dans l'état **`running`**

```
JSONPath:
$.Reservations..Instances[?(@.State.Name != 'running')].ImageId

Returns:
[
  "ami-12345678"
]

Type: StringList | String
```

Création d'intégrations webhook pour Automation

Pour envoyer des messages à l'aide de webhooks pendant une automatisation, créez une intégration. Les intégrations peuvent être appelées lors d'une automatisation à l'aide de l'action `aws:invokeWebhook` dans votre runbook. Si vous n'avez pas encore créé de webhook, consultez Créer des webhooks pour les intégrations. Pour en savoir plus sur l'action `aws:invokeWebhook`, consultez `aws:invokeWebhook` : appeler une intégration de webhook Automation.

Comme indiqué dans les procédures suivantes, vous pouvez créer une intégration à l'aide de la console Systems Manager Automation ou de l'outil de ligne de commande de votre choix.

Créer des intégrations (console)

Pour créer une intégration pour Automation (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation de gauche, sélectionnez Automation (Automatisation).
3. Sélectionnez l'onglet Integrations (Intégrations).
4. Sélectionnez Add integration (Ajouter une intégration), puis Webhook.
5. Saisissez les valeurs requises et les valeurs facultatives que vous souhaitez inclure pour l'intégration.
6. Sélectionnez Add (Ajouter) pour créer l'intégration.

Créer des intégrations (ligne de commande)

Pour créer une intégration à l'aide des outils de ligne de commande, vous devez créer le paramètre `SecureString` requis pour une intégration. Automation utilise un espace de noms réservé dans Parameter Store, une fonctionnalité de Systems Manager, pour stocker des informations sur votre intégration. Si vous créez une intégration à l'aide de la AWS Management Console, Automation gère ce processus à votre place. Après l'espace de noms, vous devez spécifier le type d'intégration que vous souhaitez créer, puis le nom de votre intégration. Automation prend actuellement en charge les intégrations de type `webhook`.

Les champs pris en charge pour les intégrations de type `webhook` sont les suivants :

- Description
- headers
- payload
- URL

Avant de commencer

Si vous ne l'avez pas déjà fait, installez et configurez l'AWS Command Line Interface (AWS CLI) ou les AWS Tools for PowerShell. Pour plus d'informations, consultez la section Installation ou mise à jour de la version la plus récente de l'AWS CLI et Installation d'AWS Tools for PowerShell.

Pour créer une intégration pour Automation (ligne de commande)

- Exécutez les commandes suivantes pour créer le paramètre `SecureString` requis pour une intégration. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations. L'espace de noms `/d9d01087-4a3f-49e0-b0b4-d568d7826553/ssm/integrations/webhook/` est réservé dans Parameter Store pour les intégrations. Le nom de votre paramètre doit utiliser cet espace de noms suivi du nom de votre intégration. Par exemple `/d9d01087-4a3f-49e0-b0b4-d568d7826553/ssm/integrations/webhook/`*`myWebhookIntegration`*.

  Linux & macOS

  ```
  aws ssm put-parameter \
      --name "/d9d01087-4a3f-49e0-b0b4-d568d7826553/ssm/integrations/
  webhook/myWebhookIntegration" \
      --type "SecureString" \
      --data-type "aws:ssm:integration" \
      --value '{"description": "My first webhook integration for Automation.",
   "url": "myWebHookURL"}'
  ```

  Windows

  ```
  aws ssm put-parameter ^
      --name "/d9d01087-4a3f-49e0-b0b4-d568d7826553/ssm/integrations/
  webhook/myWebhookIntegration" ^
      --type "SecureString" ^
      --data-type "aws:ssm:integration" ^
      --value  "{\"description\":\"My first webhook integration for Automation.\",
  \"url\":\"myWebHookURL\"}"
  ```

  PowerShell

  ```
  Write-SSMParameter `
      -Name "/d9d01087-4a3f-49e0-b0b4-d568d7826553/ssm/integrations/
  webhook/myWebhookIntegration" `
      -Type "SecureString"
      -DataType "aws:ssm:integration"
      -Value '{"description": "My first webhook integration for Automation.",
   "url": "myWebHookURL"}'
  ```

Créer des webhooks pour les intégrations

Lorsque vous créez des webhooks avec votre fournisseur, prenez note des éléments suivants :

- Le protocole doit être HTTPS.
- Les en-têtes de demande personnalisés sont pris en charge.
- Un corps de requête par défaut peut être spécifié.
- Le corps de requête par défaut peut être remplacé lorsqu'une intégration est appelée à l'aide de l'action `aws:invokeWebhook`.

Gestion de délais d'expiration dans des runbooks

La propriété `timeoutSeconds` est partagée par toutes les actions de l'automatisation. 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 l'automatisation 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 que l'automatisation passe à une autre action sans affecter le statut global de l'automatisation 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 l'automatisation doit passer si un délai d'expiration a été spécifié. Spécifiez une valeur au format `step:`*`step name`* pour la propriété `onFailure` plutôt que la valeur par défaut `Abort`. Par défaut, si une action arrive à expiration, le statut de l'automatisation sera `Timed Out`. Pour empêcher un délai d'expiration d'affecter le statut d'exécution de l'automatisation, spécifiez `false` pour la propriété `isCritical`.

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 de l'automatisation, veuillez consulter Propriétés partagées par toutes les actions.

Référence du runbook Systems Manager Automation

Pour vous aider à démarrer rapidement, AWS Systems Manager fournit des runbooks prédéfinis. Ces runbooks sont maintenus par Amazon Web Services, AWS Support et AWS Config. La référence du runbook décrit chacun des runbooks prédéfinis fournis par Systems Manager, AWS Support et AWS Config. Pour de plus amples informations, veuillez consulter Référence du runbook Systems Manager Automation.

Didacticiels

Les didacticiels suivants vous aident à utiliser AWS Systems Manager Automation pour répondre à des cas d'utilisation courants. Ces didacticiels expliquent comment utiliser vos propres runbooks, des runbooks prédéfinis fournis par Automation et d'autres fonctionnalités de Systems Manager avec d'autres Services AWS.

Table des matières

        - Octroi d'autorisations à AWSSupport -EC2Rescue pour effectuer des actions sur vos instances
            - Attribution des autorisations en utilisant les politiques IAM
            - Octroi d'autorisations à l'aide d'un AWS CloudFormation modèle
    - Exécution d'Automation
- Transmission de données à Automation à l'aide de transformateurs en entrée

Mise à jour des AMIs

Les didacticiels suivants expliquent comment mettre à jour Amazon Machine Image (AMIs) pour inclure les derniers correctifs.

Rubriques

- Mettre à jour une AMI Linux
- Mettre à jour une AMI Linux (AWS CLI)
- Mettre à jour une AMI Windows Server
- Mettez à jour un golden AMI à l'aide de l'automatisation AWS Lambda, et Parameter Store
- Mise à jour AMIs grâce à l'automatisation et Jenkins
- Mise à jour d'AMIs pour des groupes Auto Scaling

Mettre à jour une AMI Linux

Cette procédure Systems Manager Automation vous montre comment utiliser la console ou l' AWS CLI et le runbook `AWS-UpdateLinuxAmi` pour mettre à jour une AMI Linux avec les correctifs les plus récents des packages que vous spécifiez. Automation est une fonctionnalité de AWS Systems Manager. Le runbook `AWS-UpdateLinuxAmi` automatise aussi l'installation de packages et configurations supplémentaires propres aux sites. Vous pouvez mettre à jour diverses distributions Linux à l'aide de cette procédure pas à pasUbuntu Server, notamment CentOS, RHEL, SLES ou Amazon Linux. AMIs Pour obtenir la liste complète des versions Linux prises en charge, consultez Conditions préalables requises Patch Manager.

Le runbook `AWS-UpdateLinuxAmi` vous permet d'automatiser les tâches de maintenance d'images sans devoir créer de runbook en JSON ou YAML. Vous pouvez utiliser le runbook `AWS-UpdateLinuxAmi` pour effectuer les types de tâche suivants.

- Mettre à niveau tous les packages de distribution et les logiciels Amazon sur une Amazon Linux, Red Hat Enterprise Linux, Ubuntu Server, SUSE Linux Enterprise Server ou CentOS Amazon Machine Image (AMI). Il s'agit du comportement du runbook par défaut.
- Effectuez l'installation AWS Systems Manager SSM Agent sur une image existante pour activer les fonctionnalités de Systems Manager, telles que l'exécution de commandes à distance AWS Systems Manager Run Command ou la collecte d'un inventaire logiciel à l'aide d'Inventory.
- Installer des packages logiciels supplémentaires

Avant de commencer

Avant de commencer à travailler avec des runbooks, configurez les rôles et, éventuellement, EventBridge pour Automation. Pour plus d'informations, consultez Configuration d'Automation. Cette procédure pas à pas nécessite également que vous spécifiiez 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 Configurer les autorisations d'instance requises pour Systems Manager.

Le runbook `AWS-UpdateLinuxAmi` accepte les paramètres d'entrée suivants.

Paramètre	Type	Description
SourceAmiId	Chaîne	(Obligatoire) L'ID d'AMI source.
IamInstanceProfileName	Chaîne	(Obligatoire) Nom du rôle de profil d'instance IAM que vous avez créé dans Configure les autorisations d'instance requises pour Systems Manager. 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 runbook utilise uniquement le nom du rôle de profil d'instance. Si vous

Paramètre	Type	Description
		spécifiez l'Amazon Resource Name (ARN), l'automatisation échoue.
AutomationAssumeRôle	Chaîne	(Obligatoire) Le nom du rôle de service IAM que vous avez créé dans Configuration d'Automation. 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 runbook. Pour ce paramètre, l'ARN complet doit être spécifié.
TargetAmiNom	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	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.

Paramètre	Type	Description
PreUpdateScénario	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.
PostUpdateScénario	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	Chaîne	(Facultatif) Mettre à jour uniquement ces packages nommés. Par défaut (\"all \"), toutes les mises à jour disponibles sont appliquées.
ExcludePackages	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 runbook `AWS-UpdateLinuxAmi` inclut les actions d'automatisation suivantes par défaut.

Étape 1 : launchInstance (action **`aws:runInstances`**)

Cette étape lance une instance à l'aide de données utilisateur Amazon Elastic Compute Cloud (Amazon EC2) et d'un rôle de profil d'instance IAM. Userdata installe l'SSM Agent approprié selon le système d'exploitation. L'installation de SSM Agent vous permet d'utiliser des fonctionnalités de Systems Manager comme Run Command, State Manager et Inventory.

Étape 2 : updateOSSoftware (action **aws:runCommand**)

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. Par exemple : « AMI Généré par EC2 Automation le {{global:Date_Time}} à partir de {{SourceAmiId}} » où DATE_TIME et sourceId représentent des variables d'automatisation.

Étape 5 : terminateInstance (action **aws:changeInstanceState**)

Cette étape nettoie l'automatisation en mettant hors service l'instance en cours d'exécution.

Sortie

L'automatisation renvoie le nouvel ID AMI en tant que sortie.

> **Note**
>
> Par défaut, lorsqu'Automation exécute le runbook `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 runbook `AWS-UpdateLinuxAmi` et spécifier un ID de sous-réseau. Pour plus d'informations, consultez VPC non défini 400.

Pour créer une AMI à laquelle un correctif a été appliqué l'aide d'Automation (AWS Systems Manager)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation de gauche, sélectionnez Automation (Automatisation).
3. Sélectionnez Execute automation (Exécuter l'automatisation).
4. Dans la liste Document Type (Type de document), sélectionnez **`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. Sélectionnez Suivant.
7. Dans la section Execution mode (Mode d'exécution), sélectionnez Simple Execution (Exécution simple).
8. Dans la section Input parameters (Paramètres d'entrée), saisissez les informations que vous avez collectées dans la section Before you begin (Avant de commencer).
9. Sélectionnez Execute (Exécuter). La console affiche le statut de l'exécution d'automatisation.

Lorsque l'automatisation est terminée, lancez une instance test à partir de l'AMI mise à jour pour vérifier les modifications.

> **Note**
>
> Si une étape de l'automatisation échoue, les informations concernant l'échec sont répertoriées dans la page Automation Executions. L'automatisation est conçue 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.

Mettre à jour une AMI Linux (AWS CLI)

Cette procédure AWS Systems Manager d'automatisation explique comment utiliser le runbook AWS Command Line Interface (AWS CLI) et le `AWS-UpdateLinuxAmi` runbook Systems Manager pour patcher automatiquement un Linux Amazon Machine Image (AMI) avec les dernières versions des packages que vous spécifiez. L'automatisation est une capacité de AWS Systems Manager. Le runbook `AWS-UpdateLinuxAmi` automatise aussi l'installation de packages et configurations supplémentaires propres aux sites. Vous pouvez mettre à jour diverses distributions Linux à l'aide de cette procédure pas à pasUbuntu Server, notamment CentOS, RHEL, SLES ou Amazon Linux. AMIs Pour obtenir la liste complète des versions Linux prises en charge, consultez Conditions préalables requises Patch Manager.

Le runbook `AWS-UpdateLinuxAmi` vous permet d'automatiser les tâches de maintenance d'images sans devoir créer de runbook en JSON ou YAML. Vous pouvez utiliser le runbook `AWS-UpdateLinuxAmi` pour effectuer les types de tâche suivants.

- Mettez à niveau tous les packages de distribution et les logiciels Amazon sur un système d'exploitation Amazon Linux Red Hat Enterprise LinuxUbuntu Server, SLES ou Cent Amazon Machine Image (AMI). Il s'agit du comportement du runbook par défaut.
- Effectuez l'installation AWS Systems Manager SSM Agent sur une image existante pour activer les fonctionnalités de Systems Manager, telles que l'exécution de commandes à distance AWS Systems Manager Run Command ou la collecte d'un inventaire logiciel à l'aide d'Inventory.
- Installer des packages logiciels supplémentaires

Avant de commencer

Avant de commencer à travailler avec des runbooks, configurez les rôles et, éventuellement, EventBridge pour Automation. Pour plus d'informations, consultez Configuration d'Automation. Cette procédure pas à pas nécessite également que vous spécifiiez 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 Configurer les autorisations d'instance requises pour Systems Manager.

Le runbook `AWS-UpdateLinuxAmi` accepte les paramètres d'entrée suivants.

Paramètre	Type	Description
SourceAmiId	Chaîne	(Obligatoire) L'ID d'AMI source. Vous pouvez référencer automatiquement le dernier identifiant d'un Amazon EC2 AMI pour Linux à l'aide d'un paramètre AWS Systems Manager Parameter Store public. Pour plus d'informations, consultez la section Rechercher les derniers AMI identifiants Amazon Linux à l'aide de AWS Systems ManagerParameter Store.
IamInstanceProfileName	Chaîne	(Obligatoire) Nom du rôle de profil d'instance IAM que vous avez créé dans Configure les autorisations d'instance requises pour Systems Manager. 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 runbook utilise uniquement le nom du rôle de profil d'instance.
AutomationAssumeRôle	Chaîne	(Obligatoire) Le nom du rôle de service IAM que vous avez créé dans Configuration d'Automation. Le rôle de service (également appelé

Paramètre	Type	Description
		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'exécuti on de l'action `aws:creat eImage` dans un runbook. Pour ce paramètre, l'ARN complet doit être spécifié.
TargetAmiNom	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	Chaîne	(Facultatif) Type d'instanc e 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.
PreUpdateScénario	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.

Paramètre	Type	Description
PostUpdateScénario	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	Chaîne	(Facultatif) Mettre à jour uniquement ces packages nommés. Par défaut (\"all \"), toutes les mises à jour disponibles sont appliquées.
ExcludePackages	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 runbook `AWS-UpdateLinuxAmi` inclut les étapes suivantes par défaut.

Étape 1 : launchInstance (action **`aws:runInstances`**)

Cette étape lance une instance à l'aide de données utilisateur Amazon Elastic Compute Cloud (Amazon EC2) et d'un rôle de profil d'instance IAM. User data installe le SSM Agent approprié selon le système d'exploitation. L'installation de SSM Agent vous permet d'utiliser des fonctionnalités de Systems Manager comme Run Command, State Manager et Inventory.

Étape 2 : updateOSSoftware (action **`aws:runCommand`**)

Cette étape exécute les commandes suivantes sur l'instance lancée :

- Télécharge un script de mise à jour à partir d'Amazon Simple Storage Service (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. Par exemple : « AMI générée par EC2 Automation le {{global:Date_time}} à partir de {{Id}} » où DATE_TIME et SourceAmi sourceId représentent les variables d'automatisation.

Étape 5 : terminateInstance (action **`aws:changeInstanceState`**)

Cette étape nettoie l'automatisation en mettant hors service l'instance en cours d'exécution.

Sortie

L'automatisation renvoie le nouvel ID AMI en tant que sortie.

> **Note**
>
> Par défaut, lorsqu'Automation exécute le runbook `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 runbook `AWS-UpdateLinuxAmi` et spécifier un ID de sous-réseau. Pour plus d'informations, consultez VPC non défini 400.

Pour créer une AMI corrigée à l'aide d'Automation

1. Installez et configurez le AWS Command Line Interface (AWS CLI), si ce n'est pas déjà fait.

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l' AWS CLI.

2. Exécutez la commande suivante pour exécuter le runbook `AWS-UpdateLinuxAmi`. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations.

   ```
   aws ssm start-automation-execution \
       --document-name "AWS-UpdateLinuxAmi" \
       --parameters \
       SourceAmiId=AMI ID, \
       IamInstanceProfileName=IAM instance profile, \
       AutomationAssumeRole='arn:aws:iam::
   {{global:ACCOUNT_ID}}:role/AutomationServiceRole'
   ```

   La commande renvoie un ID d'exécution. Copiez cet ID dans le Presse-papiers. Vous utiliserez cet ID pour afficher le statut de l'automatisation.

   ```
   {
       "AutomationExecutionId": "automation execution ID"
   }
   ```

3. Pour visualiser l'automatisation à l'aide de AWS CLI, exécutez la commande suivante :

   ```
   aws ssm describe-automation-executions
   ```

4. Pour afficher les détails de la progression de l'automatisation, exécutez la commande suivante. Remplacez *`automation execution ID`* (ID d'exécution de l'automatisation) par vos propres informations.

   ```
   aws ssm get-automation-execution --automation-execution-id automation execution ID
   ```

Le processus de mise à jour peut prendre 30 minutes ou plus.

> **Note**
>
> Vous pouvez aussi surveiller le statut de l'automatisation dans la console. Dans la liste, sélectionnez l'automatisation que vous venez juste d'exécuter, puis sélectionnez l'onglet Steps (Étapes). Cet onglet affiche l'état des actions de l'automatisation.

Lorsque l'automatisation est terminée, lancez une instance test à partir de l'AMI mise à jour pour vérifier les modifications.

> **Note**
>
> Si une étape de l'automatisation échoue, les informations concernant l'échec sont répertoriées dans la page Automation Executions. L'automatisation est conçue 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.

Mettre à jour une AMI Windows Server

Le runbook `AWS-UpdateWindowsAmi` vous permet d'automatiser les tâches de maintenance d'images sur vos Amazon Machine Image (AMI) Windows Amazon sans devoir créer le flux de travail en JSON ou YAML. Ce runbook est pris en charge pour Windows Server 2008 R2 ou une version ultérieure. Vous pouvez utiliser le runbook `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 des runbooks, configurez des rôles pour Automation afin d'ajouter une politique `iam:PassRole` qui référence l'ARN du profil d'instance auquel vous souhaitez octroyer l'accès. Vous pouvez éventuellement configurer Amazon EventBridge for Automation, une fonctionnalité de AWS Systems Manager. Pour plus d'informations, consultez Configuration d'Automation. Cette procédure pas à pas nécessite également que vous spécifiiez 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 Configurer les autorisations d'instance requises pour Systems Manager.

> **Note**
>
> Les mises à jour de AWS Systems Manager SSM Agent sont généralement déployées dans différentes régions, à des heures différentes. Lorsque vous personnalisez ou mettez à jour une AMI, utilisez uniquement les AMI source publiées pour la région dans laquelle vous travaillez. Vous aurez ainsi l'assurance de travailler avec le dernier agent SSM Agent publié pour cette région, et vous éviterez des problèmes de compatibilité.

Le runbook `AWS-UpdateWindowsAmi` accepte les paramètres d'entrée suivants.

Paramètre	Type	Description
SourceAmiId	Chaîne	(Obligatoire) L'ID d'AMI source. Vous pouvez référencer automatiquement le dernier ID d'AMI Windows Server à l'aide d'un paramètre Systems Manager public Parameter Store. Pour plus d'informations, consultez Requête portant sur les derniers ID d'AMI Windows à l'aide de AWS Systems ManagerParameter Store.
SubnetId	Chaîne	(Facultatif) Le sous-réseau dans lequel vous souhaitez lancer l'instance temporair

Paramètre	Type	Description
		e. Vous devez spécifier une valeur pour ce paramètre si vous avez supprimé votre VPC par défaut.
IamInstanceProfileName	Chaîne	(Obligatoire) Nom du rôle de profil d'instance IAM que vous avez créé dans Configure les autorisations d'instance requises pour Systems Manager. 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 runbook utilise uniquement le nom du rôle de profil d'instance.

Paramètre	Type	Description
AutomationAssumeRôle	Chaîne	(Obligatoire) Le nom du rôle de service IAM que vous avez créé dans Configuration d'Automation. 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 runbook. Pour ce paramètre, l'ARN complet doit être spécifié.
TargetAmiNom	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	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.

Paramètre	Type	Description
PreUpdateScénario	Chaîne	(Facultatif) Un script à exécuter avant la mise à jour de l'AMI. Saisissez un script dans le runbook ou lors de l'exécution en tant que paramètre.
PostUpdateScénario	Chaîne	(Facultatif) Un script à exécuter après la mise à jour de l'AMI. Saisissez un script dans le runbook ou lors de l'exécution en tant que paramètre.
IncludeKbs	Chaîne	(Facultatif) Spécifiez un ou plusieurs ID d'articles Microsoft Knowledge Base (KB) à inclure. Vous pouvez installer plusieurs ID en utilisant des valeurs séparées par une virgule. Formats valides : KB9876543 ou 9876543.
ExcludeKbs	Chaîne	(Facultatif) Spécifiez un ou plusieurs ID d'articles Microsoft Knowledge Base (KB) à exclure. Vous pouvez exclure plusieurs ID en utilisant des valeurs séparées par une virgule. Formats valides : KB9876543 ou 9876543.

Paramètre	Type	Description
Catégories	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 la sécurité, Mise à jour de la définition, Report de mise à jour, Service Pack, Outil, Mise à jour, ou Pilote. Parmi les formats valides, on compte une seule entrée comme : Critical Update. Ou, vous pouvez spécifier une liste avec des éléments séparés par des virgules : Mise à jour critique, Mise à jour de la sécurité, Mise à jour de la définition.

Paramètre	Type	Description
SeverityLevels	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 précisé. Parmi les formats valides, on compte une seule entrée comme : Critique. Ou, vous pouvez spécifier une liste avec des éléments séparés par des virgules : Critique, Important, Faible.

Étapes d'Automation

Le runbook `AWS-UpdateWindowsAmi` inclut les étapes 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 : runPreUpdate Script (**`aws:runCommand`**action)

Cette étape vous permet de spécifier un script comme chaîne exécutée avant l'installation des mises à jour.

Étape 3 : updateEC2Config (action **`aws:runCommand`**)

Cette étape utilise le `AWS-InstallPowerShellModule` runbook pour télécharger un PowerShell module AWS public. Systems Manager vérifie l'intégrité du module via un hachage SHA-256. Systems Manager vérifie ensuite le système d'exploitation afin de déterminer s'il faut mettre à jour EC2Config ou EC2Launch. EC2Config s'exécute sur Windows Server 2008 R2 via Windows Server 2012 R2. EC2Launch s'exécute sur Windows Server 2016.

Étape 4 : updateSSMAgent (action **aws:runCommand**)

Cette étape met à jour SSM Agent à l'aide du runbook `AWS-UpdateSSMAgent`.

Étape 5 : mise à jour AWSPVDriver (**aws:runCommand**action)

Cette étape met à jour les pilotes AWS PV à l'aide du `AWS-ConfigureAWSPackage` runbook.

Étape 6 : updateAwsEna NetworkDriver (**aws:runCommand**action)

Cette étape met à jour les pilotes réseau AWS ENA à l'aide du `AWS-ConfigureAWSPackage` runbook.

Étape 7 : installWindowsUpdates (**aws:runCommand**action)

Cette étape installe les mises à jour Windows à l'aide du runbook `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`, `Categories`, ou `SeverityLevels`.

Étape 8 : runPostUpdate Script (**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 `AWS-InstallPowerShellModule` runbook pour télécharger un PowerShell module AWS public. Systems Manager vérifie l'intégrité du module via un hachage SHA-256. Systems Manager exécute ensuite sysprep à l'aide de méthodes AWS prises en charge pour EC2Launch (Windows Server 2016) ou EC2Config (Windows Server 2008 R2 à 2012 R2).

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. Par exemple : « AMI générée par EC2 Automation le {{global:Date_time}} à partir de {{Id}} » où DATE_TIME et SourceAmi sourceId représentent les variables d'automatisation.

Étape 12 : TerminateInstance (**aws:changeInstanceState**action)

Cette étape nettoie l'automatisation en mettant hors service l'instance en cours d'exécution.

Sortie

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'automatisation.

> **Note**
>
> Par défaut, lorsqu'Automation exécute le runbook `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 runbook `AWS-UpdateWindowsAmi` et spécifier un ID de sous-réseau. Pour plus d'informations, consultez VPC non défini 400.

Pour créer une AMI Windows corrigée à l'aide d'Automation

1. Installez et configurez le AWS Command Line Interface (AWS CLI), si ce n'est pas déjà fait.

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l' AWS CLI.
2. Exécutez la commande suivante pour exécuter le runbook `AWS-UpdateWindowsAmi`. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations. 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 plus d'une fois, vous devez spécifier une valeur unique pour `targetAMIname`. Les noms des AMI doivent être uniques.

   ```
   aws ssm start-automation-execution \
       --document-name="AWS-UpdateWindowsAmi" \
       --parameters SourceAmiId='AMI ID',IamInstanceProfileName='IAM
    instance profile',AutomationAssumeRole='arn:aws:iam::
   {{global:ACCOUNT_ID}}:role/AutomationServiceRole'
   ```

   La commande renvoie un ID d'exécution. Copiez cet ID dans le Presse-papiers. Vous utiliserez cet ID pour afficher le statut de l'automatisation.

```
{
    "AutomationExecutionId": "automation execution ID"
}
```

3. Pour visualiser l'automatisation à l'aide de AWS CLI, exécutez la commande suivante :

```
aws ssm describe-automation-executions
```

4. Pour afficher les détails de la progression de l'automatisation, exécutez la commande suivante.

```
aws ssm get-automation-execution
    --automation-execution-id automation execution ID
```

> **Note**
>
> En fonction du nombre de correctifs appliqués, le processus de mise à jour corrective Windows exécuté dans cet exemple d'automatisation peut prendre 30 minutes ou plus pour se terminer.

Mettez à jour un golden AMI à l'aide de l'automatisation AWS Lambda, et Parameter Store

L'exemple suivant utilise le modèle où une organisation gère et applique périodiquement des correctifs sur leurs propres AMIs propriétaires plutôt que de s'appuyer sur des AMIs Amazon Elastic Compute Cloud (Amazon EC2).

La procédure suivante montre comment appliquer automatiquement des correctifs de système d'exploitation (OS) à un correctif AMI qui est déjà considéré comme le plus récent up-to-date ou le plus récentAMI. Dans l'exemple, la valeur par défaut du paramètre `SourceAmiId` est définie par un AWS Systems Manager Parameter Store paramètre appelé`latestAmi`. La valeur de `latestAmi` est mise à jour par une AWS Lambda fonction invoquée à la fin de l'automatisation. Grâce à ce processus d'automatisation, le temps et les efforts consacrés à l'application des correctifs AMIs sont réduits au minimum, car les correctifs sont toujours appliqués au maximum. up-to-date AMI Parameter Storeet l'automatisation sont des capacités de AWS Systems Manager.

Avant de commencer

Configurez les rôles d'automatisation et, éventuellement, Amazon EventBridge for Automation. Pour plus d'informations, consultez Configuration d'Automation.

Table des matières

Tâche 1 : créer un paramètre dans Systems Manager Parameter Store

Créez un paramètre dans Parameter Store qui utilise les informations suivantes :

- Nom : `latestAmi`.
- Valeur : Un ID d'AMI. Par exemple : `ami-188d6e0e`.

Pour de plus amples informations sur la création d'un paramètre de chaîne Parameter Store, consultez Création de paramètres Systems Manager.

Tâche 2 : Créer un rôle IAM pour AWS Lambda

Utilisez la procédure suivante pour créer un rôle de service IAM pour AWS Lambda. Ces politiques 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 console IAM AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le volet de navigation, sélectionnez Politiques, puis Créer une politique.
3. Sélectionnez l'onglet JSON.
4. Remplacez les contenus par défaut par la politique suivante. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations.

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

```
            "Effect": "Allow",
            "Action": "logs:CreateLogGroup",
            "Resource": "arn:aws:logs:region:123456789012:*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogStream",
                "logs:PutLogEvents"
            ],
            "Resource": [
                "arn:aws:logs:region:123456789012:log-group:/aws/lambda/function
 name:*"
            ]
        }
    ]
}
```

5. Choisissez Suivant : Balises.
6. (Facultatif) Ajoutez une ou plusieurs paires clé-valeur d'identification afin d'organiser, de suivre ou de contrôler l'accès pour cette politique.
7. Choisissez Suivant : vérification.
8. Sur la page Examiner une politique, dans le champ Nom, saisissez un nom pour la politique en ligne, tel que **`amiLambda`**.
9. Sélectionnez Créer une politique.
10. Répétez les étapes 2 et 3.
11. Collez la politique suivante. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ssm:PutParameter",
            "Resource": "arn:aws:ssm:region:123456789012:parameter/latestAmi"
        },
        {
            "Effect": "Allow",
            "Action": "ssm:DescribeParameters",
```

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

12. Choisissez Suivant : Balises.
13. (Facultatif) Ajoutez une ou plusieurs paires clé-valeur d'identification afin d'organiser, de suivre ou de contrôler l'accès pour cette politique.
14. Choisissez Suivant : vérification.
15. Sur la page Examiner une politique, dans le champ Nom, saisissez un nom pour la politique en ligne, tel que **`amiParameter`**.
16. Sélectionnez Créer une politique.
17. Dans le volet de navigation, sélectionnez Rôles, puis Créer un rôle.
18. Directement sous Cas d'utilisation, choisissez Lambda, puis Suivant.
19. Sur la page Ajouter des autorisations utilisez le champ Recherche pour localiser les deux politiques que vous avez créées précédemment.
20. Cochez la case à côté des politiques, puis sélectionnez Suivant.
21. Pour Role name (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.

22. (Facultatif) Ajoutez une ou plusieurs paires clé-valeur de balise afin d'organiser, de suivre ou de contrôler l'accès pour ce rôle, puis sélectionnez Créer le rôle.

Tâche 3 : Création d'une fonction AWS Lambda

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 Lambda console AWS Management Console et ouvrez-la à l'[adresse https://console.aws.amazon.com/lambda/](https://console.aws.amazon.com/lambda/).

2. Choisissez Créer une fonction.
3. Sur la page Create function, sélectionnez Author from scratch.
4. Sous Nom de la fonction, saisissez **`Automation-UpdateSsmParam`**.
5. Pour l'environnement d'exécution, choisissez Python 3.8.
6. Dans Architecture, sélectionnez le type de processeur informatique que Lambda doit utiliser pour exécuter la fonction, x86_64 ou arm64,
7. Sous Autorisations, développez Modifier le rôle d'exécution par défaut.
8. Sélectionnez Use an existing role (Utiliser un rôle existant), puis le rôle de service Lambda que vous avez créé lors de la tâche 2.
9. Choisissez Créer une fonction.
10. Dans la section Code source, dans l'onglet lambda_function, supprimez le code prérempli dans le champ, 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
        )

    responseString = 'Updated parameter %s with value %s.' %
 (event['parameterName'], event['parameterValue'])

    return responseString
```

11. Choisissez Fichier, Enregistrer.
12. Pour tester la fonction Lambda, dans le menu Test, sélectionnez Configurer des événements de test.
13. Pour Event name (Nom d'événement), saisissez un nom pour l'événement de test, tel que **`MyTestEvent`**.
14. Remplacez le texte existant par le code JSON suivant. Remplacez *`AMI ID`* (ID de l'AMI) avec vos propres informations pour définir la valeur de votre paramètre `latestAmi`.

```
{
   "parameterName":"latestAmi",
   "parameterValue":"AMI ID"
```

```
}
```

15. Choisissez Enregistrer.
16. Sélectionnez Test pour tester la fonction. Dans l'onglet Résultat de l'exécution, le statut doit être indiqué comme Réussi, avec d'autres détails concernant la mise à jour.

Tâche 4 : créer un runbook et corriger l'AMI

Utilisez la procédure suivante pour créer un runbook qui applique des correctifs à l'AMI que vous avez spécifiée pour le paramètre latestAmi. Une fois le flux de travail d'automatisation terminé, la valeur latestAmi est mise à jour avec l'ID de l'AMI nouvellement corrigée. Les automatisations suivantes utilisent l'AMI créée par la précédente exécution.

Pour créer et exécuter le runbook

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.
3. Pour Créer un document, choisissez Automation.
4. Pour Name (Nom), saisissez **`UpdateMyLatestWindowsAmi`**.
5. Sélectionnez l'onglet Éditeur, puis Modifier.
6. Choisissez OK lorsque vous y êtes invité.
7. Remplacez le contenu par défaut dans le champ Éditeur de document avec l'exemple de runbook YAML suivant.

```
---
description: Systems Manager Automation Demo - Patch AMI and Update ASG
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: ''
  SourceAMI:
    type: String
    description: The ID of the AMI you want to patch.
```

```
    default: '{{ ssm:latestAmi }}'
  SubnetId:
    type: String
    description: The ID of the subnet where the instance from the SourceAMI
 parameter is launched.
  SecurityGroupIds:
    type: StringList
    description: The IDs of the security groups to associate with the instance
 that's launched from the SourceAMI parameter.
  NewAMI:
    type: String
    description: The name of of newly patched AMI.
    default: 'patchedAMI-{{global:DATE_TIME}}'
  InstanceProfile:
    type: String
    description: The name of the IAM instance profile you want the source instance
 to use.
  SnapshotId:
    type: String
    description: (Optional) The snapshot ID to use to retrieve a patch baseline
 snapshot.
    default: ''
  RebootOption:
    type: String
    description: '(Optional) Reboot behavior after a patch Install operation. If
 you choose NoReboot and patches are installed, the instance is marked as non-
compliant until a subsequent reboot and scan.'
    allowedValues:
      - NoReboot
      - RebootIfNeeded
    default: RebootIfNeeded
  Operation:
    type: String
    description: (Optional) The update or configuration to perform on the instance.
 The system checks if patches specified in the patch baseline are installed on the
 instance. The install operation installs patches missing from the baseline.
    allowedValues:
      - Install
      - Scan
    default: Install
mainSteps:
  - name: startInstances
    action: 'aws:runInstances'
    timeoutSeconds: 1200
```

```
    maxAttempts: 1
    onFailure: Abort
    inputs:
      ImageId: '{{ SourceAMI }}'
      InstanceType: m5.large
      MinInstanceCount: 1
      MaxInstanceCount: 1
      IamInstanceProfileName: '{{ InstanceProfile }}'
      SubnetId: '{{ SubnetId }}'
      SecurityGroupIds: '{{ SecurityGroupIds }}'
  - name: verifyInstanceManaged
    action: 'aws:waitForAwsResourceProperty'
    timeoutSeconds: 600
    inputs:
      Service: ssm
      Api: DescribeInstanceInformation
      InstanceInformationFilterList:
        - key: InstanceIds
          valueSet:
            - '{{ startInstances.InstanceIds }}'
      PropertySelector: '$.InstanceInformationList[0].PingStatus'
      DesiredValues:
        - Online
    onFailure: 'step:terminateInstance'
  - name: installPatches
    action: 'aws:runCommand'
    timeoutSeconds: 7200
    onFailure: Abort
    inputs:
      DocumentName: AWS-RunPatchBaseline
      Parameters:
        SnapshotId: '{{SnapshotId}}'
        RebootOption: '{{RebootOption}}'
        Operation: '{{Operation}}'
      InstanceIds:
        - '{{ startInstances.InstanceIds }}'
  - 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: '{{ NewAMI }}'
      NoReboot: false
      ImageDescription: Patched AMI created by 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
```

8. Sélectionnez Create automation (Créer une automatisation).
9. Dans le panneau de navigation, sélectionnez Automation (Automatisation), puis Execute automation (Exécuter l'automatisation).
10. Dans la page Choose document (Choisir un document), choisissez l'onglet Owned by me (Possédé par moi).
11. Recherchez le UpdateMyLatestWindowsAmirunbook, puis sélectionnez le bouton sur la UpdateMyLatestWindowsAmicarte.
12. Sélectionnez Suivant.
13. Sélectionnez Exécution simple.
14. Spécifiez les valeurs des paramètres d'entrée.
15. Sélectionnez Execute (Exécuter).

16. Une fois l'automatisation terminée, sélectionnez Parameter Store dans le panneau de navigation et vérifiez que la nouvelle valeur de `latestAmi` correspond à celle renvoyée par l'automatisation. Vous pouvez également vérifier que l'ID de la nouvelle AMI correspond à la sortie de l'automatisation dans la section AMI de la console Amazon EC2.

Mise à jour AMIs grâce à l'automatisation et Jenkins

Si votre organisation utilise un Jenkins logiciel dans un pipeline CI/CD, vous pouvez ajouter Automation en tant qu'étape de post-construction pour préinstaller les versions de l'application dans (). Amazon Machine Images AMIs L'automatisation est une capacité de AWS Systems Manager. Vous pouvez également utiliser la fonction de Jenkins planification pour appeler Automation et créer votre propre cadence d'application des correctifs pour votre système d'exploitation (OS).

L'exemple ci-dessous montre comment invoquer Automation depuis un Jenkins serveur qui s'exécute sur site ou dans Amazon Elastic Compute Cloud (Amazon EC2). Pour l'authentification, le Jenkins serveur utilise des AWS informations d'identification basées sur une politique IAM que vous avez créée dans l'exemple et que vous associez à votre profil d'instance.

> **Note**
>
> Veillez à suivre les meilleures pratiques en matière Jenkins de sécurité lors de la configuration de votre instance.

Avant de commencer

Effectuez les tâches suivantes avant de configurer Automation avec Jenkins :

- Complétez l'exemple Mettez à jour un golden AMI à l'aide de l'automatisation AWS Lambda, et Parameter Store. L'exemple suivant utilise le UpdateMyLatestWindowsAmirunbook créé dans cet exemple.
- Configurez les rôles IAM pour Automation. Systems Manager nécessite un rôle de profil d'instance et un ARN de rôle de service pour traiter les automatisations. Pour plus d'informations, consultez Configuration d'Automation.

Pour créer une politique IAM pour le serveur Jenkins

1. Connectez-vous à la console IAM AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le volet de navigation, sélectionnez Politiques, puis Créer une politique.
3. Sélectionnez l'onglet JSON.
4. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

```
{
        "Version": "2012-10-17",
        "Statement": [
            {
                "Effect": "Allow",
                "Action": "ssm:StartAutomationExecution",
                "Resource": [
                    "arn:aws:ssm:region:account ID:document/
UpdateMyLatestWindowsAmi",
                    "arn:aws:ssm:region:account ID:automation-definition/
UpdateMyLatestWindowsAmi:$DEFAULT"
                ]
            }
        ]
    }
```

5. Sélectionnez Examiner une politique.
6. Sur la page Examiner une politique, dans le champ Nom, saisissez un nom pour la politique en ligne, tel que **`JenkinsPolicy`**.
7. Sélectionnez Créer une politique.
8. Dans le panneau de navigation, choisissez Roles (Rôles).
9. Choisissez le profil d'instance attaché à votre Jenkins serveur.
10. Sous l'onglet Autorisations, sélectionnez Ajouter des autorisations et choisissez Attacher des politiques.
11. Dans la section Autres politiques d'autorisation, saisissez le nom de la politique que vous avez créée lors des étapes précédentes. Par exemple, JenkinsPolicy.
12. Sélectionnez la case en regard de votre politique et choisissez Attacher des politiques.

Utilisez la procédure suivante pour configurer le AWS CLI sur votre Jenkins serveur.

Pour configurer le Jenkins serveur pour l'automatisation

1. Connectez-vous à votre Jenkins serveur 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 le mot de passe, exécutez la commande suivante.

   ```
   sudo cat /var/lib/jenkins/secrets/initialAdminPassword
   ```
3. Le script Jenkins d'installation vous dirige vers la Jenkins page Personnaliser. Sélectionnez Install suggested plugins (Installer les plug-ins suggérés).
4. Une fois l'installation terminée, choisissez Administrator Credentials, sélectionnez Enregistrer les informations d'identification, puis sélectionnez Commencer à utiliser Jenkins.
5. Dans le volet de navigation de gauche, choisissez Manage Jenkins, puis Manage Plugins.
6. Cliquez sur l'onglet Available (Disponible), puis saisissez **`Amazon EC2 plugin`**.
7. Sélectionnez la case à cocher pour **`Amazon EC2 plugin`**, puis sélectionnez Install without restart (Installer sans redémarrer).
8. Une fois l'installation terminée, sélectionnez Go back to the top page (Revenir à la page supérieure).
9. Choisissez Gérer Jenkins, puis sélectionnez Gérer les nœuds et les clouds.
10. Dans la section Configurer les clouds, sélectionnez Ajouter un nouveau cloud, puis Amazon EC2.
11. Saisissez vos informations dans les champs restants. Assurez-vous de sélectionner l'option Utiliser le profil d'instance EC2 pour obtenir des informations d'identification.

Utilisez la procédure suivante pour configurer votre Jenkins projet afin d'invoquer Automation.

Pour configurer votre Jenkins serveur afin d'invoquer Automation

1. Ouvrez la Jenkins console dans un navigateur Web.
2. Sélectionnez le projet que vous voulez configurer avec Automation, puis sélectionnez Configure.
3. Dans l'onglet Build, sélectionnez Add Build Step.
4. Sélectionnez Execute shell ou Execute Windows batch command (en fonction de votre système d'exploitation).

5. Dans le champ Commande, exécutez une AWS CLI commande comme suit. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

```
aws ssm start-automation-execution \
        --document-name runbook name \
        --region Région AWS of your source AMI \
        --parameters runbook parameters
```

L'exemple de commande suivant utilise le UpdateMyLatestWindowsAmirunbook et le paramètre Systems Manager `latestAmi` créés dansMettez à jour un golden AMI à l'aide de l'automatisation AWS Lambda, et Parameter Store.

```
aws ssm start-automation-execution \
        --document-name UpdateMyLatestWindowsAmi \
        --parameters \
            "sourceAMIid='{{ssm:latestAmi}}'"
        --region region
```

DansJenkins, la commande ressemble à l'exemple de la capture d'écran suivante.

6. Dans le Jenkins projet, choisissez Build Now. Jenkinsrenvoie une sortie similaire à celle de l'exemple suivant.

Mise à jour d'AMIs pour des groupes Auto Scaling

L'exemple suivant met à jour un groupe Auto Scaling 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 de l'automatisation dans cet exemple utilise une fonction Python pour créer un nouveau modèle de lancement qui utilise l'AMI nouvellement corrigée. Ensuite, le groupe Auto Scaling est mis à jour pour utiliser le nouveau modèle 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 les rôles IAM pour l'automatisation, une fonctionnalité de AWS Systems Manager. Systems Manager nécessite un rôle de profil d'instance et un ARN de rôle de service pour traiter les automatisations. Pour plus d'informations, consultez Configuration d'Automation.

Création du runbook PatchAmi ASG AndUpdate

Utilisez la procédure suivante pour créer le runbook AndUpdatePatchAmi ASG qui corrige le paramètre SourceAMI que AMI vous spécifiez. Le runbook met également à jour un groupe Auto Scaling afin d'utiliser la dernière AMI corrigée.

Pour créer et exécuter le runbook

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.
3. Dans le menu déroulant Create document (Créer un document), sélectionnez Automation.
4. Dans le champ Nom, saisissez **`PatchAMIAndUpdateASG`**.
5. Sélectionnez l'onglet Editor (Éditeur), puis cliquez sur le bouton Edit (Modifier).
6. Sélectionnez OK à l'invite et supprimez le contenu dans le champ Document editor (Éditeur de document).
7. Dans le champ Document editor (Éditeur de document), collez l'exemple de contenu de runbook YAML suivant.

```
---
description: Systems Manager Automation Demo - Patch AMI and Update ASG
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: ''
  SourceAMI:
    type: String
    description: '(Required) The ID of the AMI you want to patch.'
  SubnetId:
    type: String
    description: '(Required) The ID of the subnet where the instance from the
 SourceAMI parameter is launched.'
  SecurityGroupIds:
    type: StringList
    description: '(Required) The IDs of the security groups to associate with the
 instance launched from the SourceAMI parameter.'
  NewAMI:
    type: String
    description: '(Optional) The name of of newly patched AMI.'
    default: 'patchedAMI-{{global:DATE_TIME}}'
  TargetASG:
```

```
    type: String
    description: '(Required) The name of the Auto Scaling group you want to
 update.'
  InstanceProfile:
    type: String
    description: '(Required) The name of the IAM instance profile you want the
 source instance to use.'
  SnapshotId:
    type: String
    description: (Optional) The snapshot ID to use to retrieve a patch baseline
 snapshot.
    default: ''
  RebootOption:
    type: String
    description: '(Optional) Reboot behavior after a patch Install operation. If
 you choose NoReboot and patches are installed, the instance is marked as non-
compliant until a subsequent reboot and scan.'
    allowedValues:
      - NoReboot
      - RebootIfNeeded
    default: RebootIfNeeded
  Operation:
    type: String
    description: (Optional) The update or configuration to perform on the instance.
 The system checks if patches specified in the patch baseline are installed on the
 instance. The install operation installs patches missing from the baseline.
    allowedValues:
      - Install
      - Scan
    default: Install
mainSteps:
  - name: startInstances
    action: 'aws:runInstances'
    timeoutSeconds: 1200
    maxAttempts: 1
    onFailure: Abort
    inputs:
      ImageId: '{{ SourceAMI }}'
      InstanceType: m5.large
      MinInstanceCount: 1
      MaxInstanceCount: 1
      IamInstanceProfileName: '{{ InstanceProfile }}'
      SubnetId: '{{ SubnetId }}'
      SecurityGroupIds: '{{ SecurityGroupIds }}'
```

```
  - name: verifyInstanceManaged
    action: 'aws:waitForAwsResourceProperty'
    timeoutSeconds: 600
    inputs:
      Service: ssm
      Api: DescribeInstanceInformation
      InstanceInformationFilterList:
        - key: InstanceIds
          valueSet:
            - '{{ startInstances.InstanceIds }}'
      PropertySelector: '$.InstanceInformationList[0].PingStatus'
      DesiredValues:
        - Online
    onFailure: 'step:terminateInstance'
  - name: installPatches
    action: 'aws:runCommand'
    timeoutSeconds: 7200
    onFailure: Abort
    inputs:
      DocumentName: AWS-RunPatchBaseline
      Parameters:
        SnapshotId: '{{SnapshotId}}'
        RebootOption: '{{RebootOption}}'
        Operation: '{{Operation}}'
      InstanceIds:
        - '{{ startInstances.InstanceIds }}'
  - 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: '{{ NewAMI }}'
      NoReboot: false
      ImageDescription: Patched AMI created by Automation
  - name: terminateInstance
```

```
    action: 'aws:changeInstanceState'
    maxAttempts: 1
    onFailure: Continue
    inputs:
      InstanceIds:
        - '{{ startInstances.InstanceIds }}'
      DesiredState: terminated
  - name: updateASG
    action: 'aws:executeScript'
    timeoutSeconds: 300
    maxAttempts: 1
    onFailure: Abort
    inputs:
      Runtime: python3.8
      Handler: update_asg
      InputPayload:
        TargetASG: '{{TargetASG}}'
        NewAMI: '{{createImage.ImageId}}'
      Script: |-
        from __future__ import print_function
        import datetime
        import json
        import time
        import boto3

        # create auto scaling and ec2 client
        asg = boto3.client('autoscaling')
        ec2 = boto3.client('ec2')

        def update_asg(event, context):
            print("Received event: " + json.dumps(event, indent=2))

            target_asg = event['TargetASG']
            new_ami = event['NewAMI']

            # get object for the ASG we're going to update, filter by name of
target ASG
            asg_query =
asg.describe_auto_scaling_groups(AutoScalingGroupNames=[target_asg])
            if 'AutoScalingGroups' not in asg_query or not
asg_query['AutoScalingGroups']:
                return 'No ASG found matching the value you specified.'
```

```
            # gets details of an instance from the ASG that we'll use to model the
 new launch template after
            source_instance_id = asg_query.get('AutoScalingGroups')[0]['Instances']
[0]['InstanceId']
            instance_properties = ec2.describe_instances(
                InstanceIds=[source_instance_id]
            )
            source_instance = instance_properties['Reservations'][0]['Instances']
[0]

            # create list of security group IDs
            security_groups = []
            for group in source_instance['SecurityGroups']:
                security_groups.append(group['GroupId'])

            # create a list of dictionary objects for block device mappings
            mappings = []
            for block in source_instance['BlockDeviceMappings']:
                volume_query = ec2.describe_volumes(
                    VolumeIds=[block['Ebs']['VolumeId']]
                )
                volume_details = volume_query['Volumes']
                device_name = block['DeviceName']
                volume_size = volume_details[0]['Size']
                volume_type = volume_details[0]['VolumeType']
                device = {'DeviceName': device_name, 'Ebs': {'VolumeSize':
 volume_size, 'VolumeType': volume_type}}
                mappings.append(device)

            # create new launch template using details returned from instance in
 the ASG and specify the newly patched AMI
            time_stamp = time.time()
            time_stamp_string =
 datetime.datetime.fromtimestamp(time_stamp).strftime('%m-%d-%Y_%H-%M-%S')
            new_template_name = f'{new_ami}_{time_stamp_string}'
            try:
                ec2.create_launch_template(
                    LaunchTemplateName=new_template_name,
                    LaunchTemplateData={
                        'BlockDeviceMappings': mappings,
                        'ImageId': new_ami,
                        'InstanceType': source_instance['InstanceType'],
                        'IamInstanceProfile': {
                            'Arn': source_instance['IamInstanceProfile']['Arn']
```

```
                        },
                        'KeyName': source_instance['KeyName'],
                        'SecurityGroupIds': security_groups
                    }
                )
            except Exception as e:
                return f'Exception caught: {str(e)}'
            else:
                # update ASG to use new launch template
                asg.update_auto_scaling_group(
                    AutoScalingGroupName=target_asg,
                    LaunchTemplate={
                        'LaunchTemplateName': new_template_name
                    }
                )
                return f'Updated ASG {target_asg} with new launch template
 {new_template_name} which uses AMI {new_ami}.'
outputs:
  - createImage.ImageId
```

8. Sélectionnez Create automation (Créer une automatisation).
9. Dans le panneau de navigation, sélectionnez Automation (Automatisation), puis Execute automation (Exécuter l'automatisation).
10. Dans la page Choose document (Choisir un document), choisissez l'onglet Owned by me (Possédé par moi).
11. Recherchez le runbook PatchAmi AndUpdate ASG, puis sélectionnez le bouton dans la carte PatchAmi ASG. AndUpdate
12. Sélectionnez Suivant.
13. Sélectionnez Exécution simple.
14. Spécifiez les valeurs des paramètres d'entrée. Assurez-vous que le `SubnetId` et les `SecurityGroupIds` que vous spécifiez autorisent l'accès aux points de terminaison publics de Systems Manager ou à vos points de terminaison d'interface pour Systems Manager.
15. Sélectionnez Execute (Exécuter).
16. Une fois l'automatisation terminée, dans la console Amazon EC2, choisissez Auto Scaling, puis Launch Templates (Modèles de lancement). Vérifiez que vous voyez le nouveau modèle de lancement et qu'il utilise la nouvelle AMI.
17. Sélectionnez Auto Scaling, puis Auto Scaling Groups (Groupes Auto Scaling). Vérifiez que le groupe Auto Scaling utilise le nouveau modèle de lancement.

18. Mettez hors fonction une ou plusieurs instances dans votre groupe Auto Scaling. Des instances de remplacement seront lancées en utilisant la nouvelle AMI.

Utilisation de runbooks en libre-service AWS Support

Cette section décrit l'utilisation de certaines des automatisations en libre-service créées par l'équipe AWS Support. Ces automatisations vous aident à gérer vos ressources AWS.

Support Automation Workflows

Support Automation Workflows (SAW) sont des runbooks d'automatisation rédigés et gérés par l'équipe AWS Support. Ces runbooks vous aident à résoudre les problèmes courants liés à vos ressources AWS, à surveiller et identifier les problèmes de réseau de manière proactive, à collecter et analyser les journaux, etc.

Les runbooks SAW utilisent le préfixe **`AWSSupport`**. Par exemple, `AWSSupport-ActivateWindowsWithAmazonLicense`.

En outre, les clients AWS Enterprise and Business Support ont également accès aux runbooks utilisant le préfixe **`AWSPremiumSupport`**. Par exemple, `AWSPremiumSupport-TroubleshootEC2DiskUsage`.

Pour en savoir plus sur AWS Support, consultez Démarrer avec AWS Support.

Rubriques

- Exécuter l'outil EC2Rescue sur les instances inaccessibles
- Réinitialiser les mots de passe et les clés SSH sur les instances EC2

Exécuter l'outil EC2Rescue sur les instances inaccessibles

EC2Rescue peut vous aider à diagnostiquer et à résoudre les problèmes qui peuvent survenir sur les instances Amazon Elastic Compute Cloud (Amazon EC2) pour Linux et Windows Server. Vous pouvez exécuter l'outil manuellement, comme décrit dans Utilisation d'EC2Rescue pour Linux Server et Utilisation d'EC2Rescue pour Windows Server. Vous pouvez aussi exécuter l'outil automatiquement en utilisant Systems Manager Automation et le runbook **`AWSSupport-ExecuteEC2Rescue`**. L'automatisation est une capacité de AWS Systems Manager. Le runbook **`AWSSupport-ExecuteEC2Rescue`** est destiné à exécuter une combinaison d'actions Systems Manager, d'actions AWS CloudFormation et de fonctions Lambda qui automatisent les étapes normalement requises pour utiliser EC2Rescue.

Vous pouvez utiliser le runbook **AWSSupport-ExecuteEC2Rescue** pour dépanner et potentiellement corriger différents types de problèmes liés aux systèmes d'exploitation (SE). Les instances avec des volumes racines chiffrés ne sont pas prises en charge. Pour obtenir une liste complète, consultez les rubriques suivantes :

Windows : consultez Action de résolution dans Utilisation d'EC2Rescue pour Windows Server avec la ligne de commande.

Linux et macOS : certains modules EC2Rescue pour Linux détectent et tentent de corriger des problèmes. Pour plus d'informations, consultez la `aws-ec2rescue-linux`documentation de chaque module surGitHub.

Comment ça marche

Le dépannage d'une instance avec Automation et le runbook **AWSSupport-ExecuteEC2Rescue** fonctionne le la façon suivante :

- Vous spécifiez l'ID de l'instance inaccessible et lancez le runbook.
- 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 l'action Run Command pour exécuter EC2Rescue sur l'instance d'assistant. EC2Rescue identifie et essaye de corriger les problèmes sur le volume racine original attaché. Lorsque vous avez 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. L'instance EC2Rescue sera créée dans ce sous-réseau. Si vous ne spécifiez aucun sous-réseau, Automation crée un nouveau VPC temporaire dans votre. Compte AWS Vérifiez que vous Compte AWS disposez d'au moins un VPC. Par défaut, vous pouvez créer cinq VPC par région. Si vous avez déjà créé cinq VPC dans la région, l'automatisation échoue sans apporter les modifications à votre instance. Pour de plus amples informations sur les quotas Amazon VPC, consultez VPC et sous-réseaux dans le Guide de l'utilisateur Amazon VPC.
- Vous pouvez éventuellement créer et spécifier un rôle AWS Identity and Access Management (IAM) pour Automation. Si vous ne spécifiez pas ce rôle, Automation s'exécute dans le contexte de l'utilisateur qui a exécuté l'automatisation.

Attribution d'autorisations **`AWSSupport-EC2Rescue`** pour exécuter des actions sur vos instances

EC2Rescue a besoin d'autorisation pour réaliser une série d'action sur vos instances pendant l'automatisation. Ces actions font appel aux AWS Lambda services IAM et Amazon EC2 pour tenter de résoudre en toute sécurité les problèmes liés à vos instances. Si vous disposez d'autorisations de niveau administrateur dans votre VPC et/ou Compte AWS votre VPC, vous pourrez peut-être exécuter l'automatisation 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 politiques IAM
- Octroi d'autorisations à l'aide d'un AWS CloudFormation modèle

Attribution des autorisations en utilisant les politiques IAM

Vous pouvez attacher la politique IAM suivante à votre utilisateur, groupe ou rôle IAM en tant que politique en ligne, ou vous pouvez créer une nouvelle politique IAM gérée et la relier à votre utilisateur, groupe ou rôle. Pour plus d'informations au sujet de l'ajout d'une politique en ligne à votre compte utilisateur, groupe ou rôle, consultez la page Utilisation de politiques en ligne. Pour plus d'informations sur la création d'une nouvelle politique gérée, consultez la page Utilisation de politiques gérées.

> **Note**
>
> Si vous créez une nouvelle politique gérée par IAM, vous devez également y associer la politique AutomationRole gérée AmazonSSM afin que vos instances puissent communiquer avec l'API Systems Manager.

Politique IAM pour AWSSupport -EC2Rescue

Remplacez *account ID* (ID du compte) par vos propres informations.

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Action": [
            "lambda:InvokeFunction",
            "lambda:DeleteFunction",
            "lambda:GetFunction"
         ],
         "Resource": "arn:aws:lambda:*: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::account ID:role/AWSSupport-EC2Rescue-*",
            "arn:aws:iam::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"
      }
   ]
}
```

Octroi d'autorisations à l'aide d'un AWS CloudFormation modèle

AWS CloudFormation automatise le processus de création de rôles et de politiques IAM à l'aide d'un modèle préconfiguré. Utilisez la procédure suivante pour créer les rôles et les politiques IAM pour l'automatisation EC2Rescue à l'aide d' AWS CloudFormation.

Pour créer les rôles et les politiques IAM obligatoire pour EC2Rescue

1. Téléchargez `AWSSupport-EC2RescueRole.zip` et extrayez le fichier `AWSSupport-EC2RescueRole.json` pour l'enregistrer dans un répertoire de votre machine locale.
2. Si vous Compte AWS vous trouvez dans 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 CloudFormation console AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/cloudformation.
4. Sélectionnez Créer une pile, Avec de nouvelles ressources (standard).
5. Sur la page Créer une pile, pour Prérequis - Préparer le modèle, sélectionnez Le modèle est prêt.
6. Dans Spécifier le modèle, sélectionnez 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. Sélectionnez 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 sélectionnez 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 Next (Suivant)

12. Sur la page de révision, passez en revue les détails de la pile, puis faites défiler la page vers le bas et choisissez l'option Je reconnais que cela AWS CloudFormation pourrait créer des ressources IAM.
13. Sélectionnez Créer la pile.

    AWS CloudFormation affiche l'état de CREATE_IN_PROGRESS pendant quelques minutes. Le statut passe à CREATE_COMPLETE après la création de la pile. Vous pouvez également choisir l'icône d'actualisation pour vérifier le statut du processus de création.
14. Dans la liste Piles, sélectionnez l'option à côté de la pile que vous venez de créer, puis sélectionnez l'onglet Sorties.
15. Notez la Valeur. Il s'agit de l'ARN du AssumeRole. Vous spécifiez cet ARN lorsque vous exécutez l'automatisation lors de la procédure suivante, Exécution d'Automation.

Exécution d'Automation

> **Important**
>
> L'automatisation suivante 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 AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation de gauche, sélectionnez Automation (Automatisation).
3. Sélectionnez Execute automation (Exécuter l'automatisation).
4. Dans la section Document d'automatisation, sélectionnez M'appartenant ou appartenant à Amazon dans la liste.
5. Dans la liste des runbooks, sélectionnez le bouton de la carte pour **`AWSSupport-ExecuteEC2Rescue`**, puis sélectionnez Next (Suivant).
6. Sur la page Exécuter le document d'automatisation, sélectionnez 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 EC2 RescueInstanceType, spécifiez un type d'instance pour l'instance EC2Rescue. Le type d'instance par défaut est `t2.medium`.

   c. En AutomationAssumeRoleeffet, si vous avez créé des rôles pour cette automatisation en utilisant la AWS CloudFormation procédure décrite précédemment dans cette rubrique, choisissez l'ARN du rôle AssumeRole que vous avez créé dans la AWS CloudFormation console.

   d. (Facultatif) Pour LogDestination, spécifiez un compartiment S3 si vous souhaitez collecter des journaux au niveau du système d'exploitation lors du dépannage de 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 Default (Par défaut) la plus récente du runbook.

9. (Facultatif) Dans la section Tags (Balises), appliquez une ou plusieurs paires nom/valeur de clé de balise pour aider à identifier l'automatisation, par exemple, `Key=Purpose,Value=EC2Rescue`.

10. Sélectionnez Execute (Exécuter).

Le runbook crée une AMI de sauvegarde dans le cadre de l'automatisation. Toutes les autres ressources créées par l'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.

Réinitialiser les mots de passe et les clés SSH sur les instances EC2

Vous pouvez utiliser le runbook `AWSSupport-ResetAccess` pour réactiver automatiquement la génération du mot de passe administrateur local sur les instances Amazon Elastic Compute Cloud (Amazon EC2) pour Windows Server et générer une nouvelle clé SSH sur les instances EC2 pour Linux. Le `AWSSupport-ResetAccess` runbook est conçu pour exécuter une combinaison d' AWS Systems Manager actions, AWS CloudFormation d'actions et de AWS Lambda fonctions qui automatisent les étapes normalement requises pour réinitialiser le mot de passe de l'administrateur local.

Vous pouvez utiliser Automation, une fonctionnalité de AWS Systems Manager, avec le `AWSSupport-ResetAccess` runbook pour résoudre les problèmes suivants :

Windows

Vous avez perdu la paire de clés EC2 : pour résoudre ce problème, vous pouvez utiliser le AWSSupport- ResetAccess runbook pour créer une instance activée par mot de passe à AMI partir de votre instance actuelle, lancer une nouvelle instance depuis l'AMI et sélectionner une paire de clés qui vous appartient.

Vous avez oublié le mot de passe administrateur local : pour résoudre ce problème, vous pouvez utiliser le runbook `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 runbook `AWSSupport-ResetAccess` pour créer une clé SSH pour votre instance actuelle, ce qui vous permet de vous reconnecter à l'instance.

> **Note**
>
> Si votre instance EC2 pour Windows Server est configurée pour Systems Manager, vous pouvez aussi réinitialiser votre mot de passe administrateur local à l'aide d'EC2Rescue et de AWS Systems Manager Run Command. Pour plus d'informations, consultez la section Utilisation d'EC2Rescue pour Windows Server avec Systems Manager Run Command dans le guide de l'utilisateur Amazon EC2.

Informations connexes

Connectez-vous à votre instance Linux depuis Windows à l'aide de PuTTY dans le guide de l'utilisateur Amazon EC2

Comment ça marche

Le dépannage d'une instance avec Automation et le runbook `AWSSupport-ResetAccess` fonctionne le la façon suivante :

- Vous spécifiez l'ID de l'instance et vous exécutez le runbook.
- 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 l'action Run Command pour exécuter EC2Rescue sur l'instance d'assistant. Sous Windows, EC2Rescue active la génération de mot de passe pour l'administrateur local grâce à 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. Lorsque vous avez 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 partit de la console Amazon EC2 grâce à la paire de clés actuelle affectée à l'instance.

  Linux : vous pouvez accédez à l'instance via SSH à l'aide de la clé SSH stockée dans Systems Manager Parameter Store sous /ec2rl/openssh/***instance ID***/key (/ec2rl/openssh/ID de l'instance/key).

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. L'instance EC2Rescue sera créée dans ce sous-réseau. Si vous ne spécifiez aucun sous-réseau, Automation crée un nouveau VPC temporaire dans votre. Compte AWS Vérifiez que vous Compte AWS disposez d'au moins un VPC. Par défaut, vous pouvez créer cinq VPC par région. Si vous avez déjà créé cinq VPC dans la région, l'automatisation échoue sans apporter les modifications à votre instance. Pour de plus amples informations sur les quotas Amazon VPC, consultez VPC et sous-réseaux dans le Guide de l'utilisateur Amazon VPC.
- Vous pouvez éventuellement créer et spécifier un rôle AWS Identity and Access Management (IAM) pour Automation. Si vous ne spécifiez pas ce rôle, Automation s'exécute dans le contexte de l'utilisateur qui a exécuté l'automatisation.

Octroi d'autorisations à AWSSupport -EC2Rescue pour effectuer des actions sur vos instances

EC2Rescue a besoin d'autorisation pour réaliser une série d'action sur vos instances pendant l'automatisation. Ces actions font appel aux AWS Lambda services IAM et Amazon EC2 pour tenter de résoudre en toute sécurité les problèmes liés à vos instances. Si vous disposez d'autorisations de niveau administrateur dans votre VPC et/ou Compte AWS votre VPC, vous pourrez peut-être exécuter l'automatisation 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 politiques IAM
- Octroi d'autorisations à l'aide d'un AWS CloudFormation modèle

Attribution des autorisations en utilisant les politiques IAM

Vous pouvez attacher la politique IAM suivante à votre utilisateur, groupe ou rôle IAM en tant que politique en ligne, ou vous pouvez créer une nouvelle politique IAM gérée et la relier à votre utilisateur, groupe ou rôle. Pour plus d'informations au sujet de l'ajout d'une politique en ligne à votre compte utilisateur, groupe ou rôle, consultez la page Utilisation de politiques en ligne. Pour

plus d'informations sur la création d'une nouvelle politique gérée, consultez la page Utilisation de politiques gérées.

> **Note**
>
> Si vous créez une nouvelle politique gérée par IAM, vous devez également y associer la politique AutomationRole gérée AmazonSSM afin que vos instances puissent communiquer avec l'API Systems Manager.

Politique IAM pour **`AWSSupport-ResetAccess`**

Remplacez *`account ID`* (ID du compte) par vos propres informations.

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Action": [
            "lambda:InvokeFunction",
            "lambda:DeleteFunction",
            "lambda:GetFunction"
         ],
         "Resource": "arn:aws:lambda:*: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::account ID:role/AWSSupport-EC2Rescue-*",
            "arn:aws:iam::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"
      }
   ]
```

```
}
```

Octroi d'autorisations à l'aide d'un AWS CloudFormation modèle

AWS CloudFormation automatise le processus de création de rôles et de politiques IAM à l'aide d'un modèle préconfiguré. Utilisez la procédure suivante pour créer les rôles et les politiques IAM pour l'automatisation EC2Rescue à l'aide d' AWS CloudFormation.

Pour créer les rôles et les politiques IAM obligatoire pour EC2Rescue

1. Téléchargez `AWSSupport-EC2RescueRole.zip` et extrayez le fichier `AWSSupport-EC2RescueRole.json` pour l'enregistrer dans un répertoire de votre machine locale.
2. Si vous Compte AWS vous trouvez dans 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 CloudFormation console AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/cloudformation.
4. Sélectionnez Créer une pile, Avec de nouvelles ressources (standard).
5. Sur la page Créer une pile, pour Prérequis - Préparer le modèle, sélectionnez Le modèle est prêt.
6. Dans Spécifier le modèle, sélectionnez 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. Sélectionnez 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 sélectionnez 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 Next (Suivant)
12. Sur la page de révision, passez en revue les détails de la pile, puis faites défiler la page vers le bas et choisissez l'option Je reconnais que cela AWS CloudFormation pourrait créer des ressources IAM.
13. AWS CloudFormation affiche l'état de CREATE_IN_PROGRESS pendant quelques minutes. Le statut passe à CREATE_COMPLETE après la création de la pile. Vous pouvez également choisir l'icône d'actualisation pour vérifier le statut du processus de création.
14. Dans la liste des piles, sélectionnez l'option en regard de la pile que vous venez de créer, puis sélectionnez l'onglet Sorties.
15. Copiez la Valeur. Il s'agit de l'ARN du AssumeRole. Vous spécifierez cet ARN lorsque vous exécuterez l'automatisation.

Exécution d'Automation

La procédure suivante explique comment exécuter le runbook `AWSSupport-ResetAccess` à l'aide de la console AWS Systems Manager .

> **Important**
>
> L'automatisation 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 Run Command pour réinitialiser l'accès. Pour plus d'informations, consultez la section Utilisation d'EC2Rescue pour Windows Server avec Systems Manager Run Command dans le guide de l'utilisateur Amazon EC2.

Pour exécuter le AWSSupport - ResetAccess Automation

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation de gauche, sélectionnez Automation (Automatisation).
3. Sélectionnez Execute automation (Exécuter l'automatisation).
4. Dans la section Document d'automatisation, sélectionnez M'appartenant ou appartenant à Amazon dans la liste.

5. Dans la liste des runbooks, cliquez sur le bouton sur la carte correspondant à AWSSupport-ResetAccess, puis sur Next.
6. Sur la page Exécuter le document d'automatisation, sélectionnez 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 runbook.

   c. Pour le RescueInstancetype EC2, spécifiez un type d'instance pour l'instance EC2Rescue. Le type d'instance par défaut est `t2.medium`.

   d. En AssumeRoleeffet, si vous avez créé des rôles pour cette automatisation à l'aide de la AWS CloudFormation procédure décrite plus haut dans cette rubrique, spécifiez l' AssumeRole ARN que vous avez noté dans la AWS CloudFormation console.
9. (Facultatif) Dans la section Tags (Balises), appliquez une ou plusieurs paires nom/valeur de clé de balise pour aider à identifier l'automatisation, par exemple, `Key=Purpose,Value=ResetAccess`.
10. Sélectionnez Execute (Exécuter).
11. Pour contrôler les progrès de l'automatisation, sélectionnez l'automatisation en cours d'exécution, puis sélectionnez l'onglet Steps (Étapes). Lorsque l'automatisation est terminée, sélectionnez l'onglet Descriptions, puis sélectionnez View Output (Afficher la sortie) pour consulter les résultats. Pour consulter la sortie des étapes individuelles, sélectionnez l'onglet Étapes, puis sélectionnez Afficher les sorties en regard d'une étape.

Le runbook crée une AMI de sauvegarde et une AMI activée par mot de passe dans le cadre de l'automatisation. Toutes les autres ressources créées par le flux de travail d'automatisation sont

automatiquement supprimées, mais ces AMIs restent dans votre compte. Les AMIs sont nommées 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 de l'ID d'instance`*

Vous pouvez localiser ces 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***/key (/ec2rl/openssh/ID de l'instance/key).

Transmission de données à Automation à l'aide de transformateurs en entrée

Ce didacticiel AWS Systems Manager Automation montre comment utiliser la fonction de transformateur en entrée d'Amazon EventBridge pour extraire l'`instance-id` d'une instance Amazon Elastic Compute Cloud (Amazon EC2) à partir d'un événement de changement d'état d'instance. Automation est une fonctionnalité de AWS Systems Manager. Nous utilisons le transformateur d'entrée pour transmettre ces données à la cible du runbook `AWS-CreateImage` comme paramètre d'entrée `InstanceId`. La règle est déclenchée lorsqu'une instance passe à l'état `stopped`.

Pour de plus amples informations sur l'utilisation des transformateurs en entrée, veuillez consulter Didacticiel : Utiliser le transformateur en entrée pour personnaliser ce qui est transmis à la cible d'événement dans le Guide de l'utilisateur Amazon EventBridge.

Avant de commencer

Vérifiez que vous avez ajouté les autorisations et la politique d'approbation requises pour EventBridge à votre rôle de service Systems Manager Automation. Pour plus d'informations, consultez Présentation de la gestion des autorisations d'accès à vos ressources EventBridge dans le Guide de l'utilsiateur 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.
3. Choisissez Create rule.

4. 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.

5. Pour Event bus (Bus d'événement), sélectionnez le bus d'événement que vous souhaitez associer à cette règle. Si vous souhaitez que cette règle s'applique aux événements correspondants provenant de votre propre Compte AWS, sélectionnez défaut. Lorsqu'un Service AWS de votre compte émet un événement, il accède toujours au bus d'événement par défaut de votre compte.
6. Pour Rule type (Type de règle), choisissez Rule with an event pattern (Règle avec un modèle d'événement).
7. Choisissez Next (Suivant).
8. Pour Event source (Origine de l'événement), choisissez events or EventBridge partner events (Événements AWS ou événements partenaires EventBridge).
9. Dans la section Event pattern (Modèle d'événement), choisissez Event pattern form (Modèle d'événement).
10. Pour Event source (Origine de l'événement), choisissez AWSservices (Services).
11. Pour AWS service choisissez EC2.
12. Pour Event Type (Type d'événement), sélectionnez EC2 Instance State-change Notification (Notification de changement d'état de l'instance EC2).
13. Pour Specific state(s) (États spécifiques), choisissez stopped (arrêté).
14. Choisissez Next (Suivant).
15. Pour Types de cibles, choisissez service AWS.
16. Pour Select a target (Sélectionner une cible), choisissez Systems Manager Automation.
17. Pour Document, choisissez AWS-CreateImage.
18. Dans la section Configure automation parameter(s) (Configurer le(s) paramètre(s) d'automatisation), choisissez Input Transformer (Transformateur d'entrée).
19. Pour Input path (Chemin d'entrée), saisissez **`{"instance":"$.detail.instance-id"}`**.
20. Pour Template (Modèle), saisissez **`{"InstanceId":[<instance>]}`**.
21. Pour Execution role (Fonction d'exécution), choisissez Use existing role (Utiliser la fonction existante) et sélectionnez votre fonction du service Automation.
22. Choisissez Next (Suivant).

23. (Facultatif) Saisissez une ou plusieurs balises pour la règle. Pour plus d'informations, consultez Balisage de vos ressources Amazon EventBridge dans le Guide de l'utilisateur Amazon EventBridge.
24. Choisissez Next (Suivant).
25. Consultez les détails de la règle et choisissez Create rule (Créer une règle).

Comprendre les statuts d'automatisation

AWS Systems Manager L'automatisation fournit des informations d'état détaillées sur les différents statuts que traverse une action ou une étape d'automatisation lorsque vous exécutez une automatisation et pour l'automatisation globale. L'automatisation est une capacité de AWS Systems Manager. Vous pouvez surveiller les statuts d'automatisation à l'aide des méthodes suivantes :

- Surveillez le statut d'exécution dans la console Systems Manager Automation.
- Utilisez vos outils de ligne de commande préférés. Pour le AWS Command Line Interface (AWS CLI), vous pouvez utiliser describe-automation-step-executions ou get-automation-execution. Pour cela AWS Tools for Windows PowerShell, vous pouvez utiliser Get-SSM AutomationStep Execution ou Get-SSM. AutomationExecution
- Configurez Amazon EventBridge pour qu'il réponde aux changements de statut liés aux actions ou à l'automatisation.

Pour plus d'informations sur la gestion des délais d'expiration dans le cadre d'une automatisation, consultezGestion de délais d'expiration dans des runbooks.

À propos des statuts Automation

Automation rapporte des détails sur le statut d'actions d'automatisation individuelles, en plus de l'automatisation globale.

Le statut de l'automatisation globale peut être différent de celui rapporté par une action ou une étape individuelle, comme l'indiquent les tableaux suivants.

Statut détaillé des actions

Statut	Détails
En attente	L'exécution de l'étape n'a pas commencé. Si votre automatisation utilise des actions

Statut	Détails
	conditionnelles, les étapes restent dans cet état une fois l'automatisation terminée si la condition exigée pour l'exécution de l'étape n'est pas remplie. Les étapes restent également dans cet état si l'automatisation est annulée avant l'exécution de l'étape.
InProgress	L'étape est en cours d'exécution.
En attente	L'étape attend une entrée.
Réussite	L'étape s'est terminée avec succès. Il s'agit d'un statut de terminal.
TimedOut	Une étape ou une approbation ne s'est pas effectuée avant la période d'expiration spécifiée . Il s'agit d'un statut de terminal.
Annulation	L'étape est en cours d'arrêt après avoir été annulée par un demandeur.
Annulée	L'étape a été arrêtée par un demandeur avant d'être terminée. Il s'agit d'un statut de terminal.
Échec	L'étape ne s'est pas terminée avec succès. Il s'agit d'un statut de terminal.
Exited	Renvoyé uniquement par l'action `aws:loop`. La boucle n'était pas complètement terminée. Une étape de la boucle a été déplacée vers une étape extérieure à l'aide des propriétés `nextStep`, `onCancel` ou `onFailure` .

Statut détaillé d'une automatisation

Statut	Détails
En attente	L'exécution de l'automatisation n'a pas commencé.
InProgress	L'automatisation est en cours d'exécution.
En attente	L'automatisation attend une entrée.
Réussite	L'automatisation s'est terminée avec succès. Il s'agit d'un statut de terminal.
TimedOut	Une étape ou une approbation ne s'est pas effectuée avant la période d'expiration spécifiée . Il s'agit d'un statut de terminal.
Annulation	L'automatisation est en cours d'arrêt après avoir été annulée par un demandeur.
Annulée	L'automatisation a été arrêtée par un demandeur avant d'être terminée. Il s'agit d'un statut de terminal.
Échec	L'automatisation ne s'est pas terminée avec succès. Il s'agit d'un statut de terminal.

Résolution des problèmes liés à Systems Manager Automation

Utilisez les informations suivantes pour vous aider à résoudre les problèmes liés à AWS Systems Manager l'automatisation, une fonctionnalité de AWS Systems Manager. La rubrique inclut des tâches spécifiques pour résoudre des problèmes selon les messages d'erreur d'Automation.

Rubriques

- L'exécution a démarré, mais a expiré

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 runbooks `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 automatisation peut échouer avec une erreur de refus d'accès ou une erreur de prise de rôle non valide si vous n'avez pas correctement configuré les rôles AWS Identity and Access Management (IAM) et les politiques d'automatisation.

Accès refusé

Les exemples suivants décrivent les situations lorsqu'une automatisation a échoué lors du démarrage, avec une erreur d'accès refusé.

Accès refusé à l'API Systems Manager

Message d'erreur : `User: user arn isn't 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 : l'utilisateur tentant de démarrer l'automatisation n'est pas autorisé à appeler l'API `StartAutomationExecution`. Pour résoudre ce problème, attachez la politique IAM requise à l'utilisateur qui a été utilisé pour lancer l'automatisation.
- Cause possible 2 : l'utilisateur tentant de lancer l'automatisation dispose de l'autorisation d'appeler l'API `StartAutomationExecution`, mais n'a pas l'autorisation d'appeler l'API en utilisant le

runbook spécifique. Pour résoudre ce problème, attachez la politique IAM requise à l'utilisateur qui a été utilisé pour lancer l'automatisation.

Accès refusé en raison d' PassRole autorisations manquantes

Message d'erreur : `User: user arn isn't 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)`

L'utilisateur qui tente de démarrer l'automatisation n'est pas PassRole autorisé à assumer le rôle. Pour résoudre ce problème, associez la PassRole politique iam : au rôle de l'utilisateur qui tente de démarrer l'automatisation. Pour plus d'informations, consultez Tâche 2 : associer la PassRole politique iam : à votre rôle d'automatisation.

Rôle de responsable non valide

Lorsque vous exécutez Automation, un rôle de responsable est fourni dans le runbook ou transmis en tant que valeur de paramètre pour le runbook. 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 isn't valid.` Le rôle de responsable n'est pas mis en forme correctement. Pour résoudre ce problème, vérifiez qu'un rôle de responsable valide est spécifié dans votre runbook ou en tant que paramètre d'exécution lorsque vous lancez l'automatisation.

Supposer que le rôle ne peut pas 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 the section called "Configuration d'Automation". Des détails spécifiques pour créer ce rôle sont décrits dans la rubrique suivante, Tâche 1 : Création d'un rôle de service pour Automation.

- Cause possible 2 : le rôle de responsable n'a pas de relation d'approbation avec le service Systems Manager. Pour résoudre ce problème, créez la relation d'approbation. Pour de plus amples informations, consultez Je ne peux pas assumer un rôle dans le Guide de l'utilisateur IAM.

L'exécution a démarré, mais le statut est Failed (Échec)

Échecs spécifiques à l'action

Les runbooks contiennent des étapes, qui s'exécutent dans l'ordre. Chaque étape appelle une ou plusieurs API de Service AWS . Les API déterminent 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 Amazon Elastic Compute Cloud (Amazon EC2), sélectionnez le lien View Outputs (Afficher les sorties) de l'étape qui a échoué. Pour voir un message d'échec provenant du AWS CLI, appelez `get-automation-execution` et recherchez l'`FailureMessage`attribut dans un échec`StepExecution`.

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's launching the instance(s). Get Exception from RunInstances API of ec2 Service. Exception Message from RunInstances API: [The image id '[ami id]' doesn't 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 runbook ou les valeurs de paramètre avec l'ID d'AMI correct.

Supposons que la politique des rôles ne dispose pas d'autorisations suffisantes

Message d'erreur : `Automation Step Execution fails when it's launching the instance(s). Get Exception from RunInstances API of ec2 Service. Exception Message from RunInstances API: [You aren't authorized to perform this operation. Encoded authorization failure message: xxxxxx (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 de responsable ne dispose pas des autorisations suffisantes pour appeler l'API `RunInstances` sur des instances EC2. Pour résoudre ce problème, attachez une politique IAM au rôle assume qui a l'autorisation d'appeler l'API `RunInstances`. Pour plus d'informations, consultez le Méthode 2 : Utiliser IAM afin de configurer des rôles pour Automation.

État inattendu

Message d'erreur : `Step fails when it's 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 : un problème est survenu avec l'instance ou le service Amazon EC2. 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 pour l'action `aws:runInstances` rencontre un problème ou a 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	Services AWS invoqué par cette action	Pour obtenir des informations sur ce service	Contenu de dépannage
`aws:runInstances`	Amazon EC2	Guide de l'utilisateur Amazon EC2	Dépannage des instances EC2
`aws:changeInstanceState`	Amazon EC2	Guide de l'utilisateur Amazon EC2	Dépannage des instances EC2

Action	Services AWS invoqué par cette action	Pour obtenir des informations sur ce service	Contenu de dépannage
`aws:runCommand`	Systems Manager	AWS Systems Manager Run Command	Résolution des problèmes liés à Run Command de Systems Manager
`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
`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 Manuel du développeur	Résolution des problèmes 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 runbook 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 le paramètre `timeoutSeconds` dans l'action `aws:runInstances`. Si cela ne résout pas le problème, cherchez pourquoi l'étape est plus longue à s'exécuter que prévu.

AWS Systems Manager Change Calendar

Change Calendar, une fonctionnalité de AWS Systems Manager vous permet de configurer des plages de dates et de temps dans lesquelles les actions que vous spécifiez (dans des runbooks Systems Manager Automation, par exemple) peuvent être effectuées, ou non, dans votre Compte AWS. Dans Change Calendar, ces plages sont appelées des événements. Lorsque vous créez une entrée Change Calendar, vous créez un document Systems Manager de type `ChangeCalendar`. Dans Change Calendar, le document stocke les données iCalendar 2.0 au format texte brut. Les événements que vous ajoutez à l'entrée Change Calendar font partie du document. Pour vos premiers pas dans Change Calendar, ouvrez Systems Manager console. Dans le panneau de navigation, sélectionnez Change Calendar.

Vous pouvez créer un calendrier et ses événements dans la console Systems Manager. Vous pouvez également importer un fichier iCalendar (`.ics`), exporté à partir d'un fournisseur de calendrier tiers pris en charge, pour ajouter ses événements à votre calendrier. Les fournisseurs pris en charge comprennent Google Calendar, Microsoft Outlook et iCloud Calendar.

Une entrée Change Calendar peut être de l'un des deux types suivants :

`DEFAULT_OPEN` ou Ouvert par défaut

Toutes les actions peuvent être exécutées par défaut, sauf pendant les événements du calendrier. Pendant les événements, l'état d'un calendrier `DEFAULT_OPEN` est `CLOSED` et les événements sont bloqués.

`DEFAULT_CLOSED` ou Fermé par défaut

Toutes les actions sont bloquées par défaut, sauf pendant les événements du calendrier. Pendant les événements, l'état d'un calendrier `DEFAULT_CLOSED` est `OPEN` et les actions sont autorisées.

Vous pouvez choisir d'ajouter automatiquement tous les flux de travail d'automatisation, les fenêtres de maintenance et les associations State Manager planifiés à un calendrier. Vous pouvez également supprimer chacun de ces types de l'affichage du calendrier.

À qui est destiné Change Calendar ?

- Clients AWS qui exécutent les types d'actions suivants :
  - Créez ou exécutez des runbooks d'automatisation.
  - Créez des demandes de modification dans Change Manager.
  - Exécutez des fenêtres de maintenance.
  - Créez des associations dans State Manager.

  L'automatisation, Change Manager, Maintenance Windows et State Manager sont des fonctionnalités d'AWS Systems Manager. En intégrant ces fonctionnalités à Change Calendar, vous pouvez autoriser ou bloquer ces types d'action en fonction du statut actuel du calendrier de modifications qui leur est associé.
- Administrateurs chargés de maintenir la cohérence, la stabilité et la fonctionnalité des configurations des nœuds gérés par Systems Manager.

Avantages d'Change Calendar

Change Calendar offre entre autres les avantages suivants.

- Vérifier les modifications avant qu'elles soient appliquées

  Une entrée Change Calendar peut vous aider à vous assurer que les modifications potentiellement destructrices apportées à votre environnement sont passées en revue avant d'être appliquées.
- Appliquer les modifications uniquement pendant les heures appropriées

  Les entrées Change Calendar permettent de préserver la stabilité de votre environnement pendant les heures des événements. Par exemple, vous pouvez créer une entrée Change Calendar 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 du statut actuel ou à venir du calendrier

  Vous pouvez exécuter l'opération d'API `GetCalendarState` de Systems Manager pour afficher du statut actuel du calendrier, son état à une heure spécifique ou la prochaine fois qu'une modification de du statut du calendrier est planifiée.

- Prise en charge d'EventBridge

  Cette fonctionnalité de Systems Manager est prise en charge en tant que type d'événement dans les règles Amazon EventBridge. Pour plus d'informations, consultez Surveillance d'événements Systems Manager avec Amazon EventBridge et Référence : modèles et types d'événements Amazon EventBridge pour Systems Manager.

Rubriques

Configuration de Change Calendar

Effectuez les opérations suivantes avant d'utiliserChange Calendar, une fonctionnalité de AWS Systems Manager.

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 le AWS Command Line Interface (AWS CLI) pour obtenir des informations sur l'état des calendriers, installez la dernière version du AWS CLI sur votre ordinateur local. Pour plus d'informations sur l'installation ou la mise à niveau de l'interface CLI, consultez Installation, mise à jour et désinstallation de la AWS CLI dans le Guide de l'utilisateur AWS Command Line Interface .
AWS Tools for PowerShell	(Facultatif) Pour utiliser les outils PowerShel l pour obtenir des informations sur l'état des calendriers, installez la dernière version de Tools for PowerShell sur votre ordinateur local. Pour plus d'informations sur l'installation ou la mise à niveau des outils pour PowerShell, consultez la section Installation des AWS Tools for PowerShell dans le guide de AWS Tools for PowerShell l'utilisateur.

Configuration d'autorisations

Si votre utilisateur, votre groupe ou votre rôle dispose des autorisations d'administrateur, vous avez accès complet à Change Calendar. Si vous ne disposez pas d'autorisations d'administrateur, un administrateur doit vous les donner en affectant la stratégie gérée `AmazonSSMFullAccess` ou en affectant une stratégie qui fournit les autorisations nécessaires à votre utilisateur, votre groupe ou votre rôle.

Les autorisations suivantes sont requises pour travailler avec Change Calendar.

Les entrées Change Calendar

Pour créer, mettre à jour ou supprimer une entrée Change Calendar, y compris ajouter et supprimer des événements de l'entrée, une politique attachée à votre utilisateur, groupe ou rôle doit autoriser les actions suivantes :

- `ssm:CreateDocument`
- `ssm:DeleteDocument`
- `ssm:DescribeDocument`
- `ssm:DescribeDocumentPermission`
- `ssm:GetCalendar`
- `ssm:ListDocuments`
- `ssm:ModifyDocumentPermission`
- `ssm:PutCalendar`
- `ssm:UpdateDocument`
- `ssm:UpdateDocumentDefaultVersion`

État du calendrier

Pour obtenir des informations sur l'état actuel ou à venir du calendrier, une politique attachée à votre utilisateur, votre groupe ou votre rôle doit autoriser l'action suivante :

- `ssm:GetCalendarState`

Événements opérationnels

Pour afficher les événements opérationnels, tels que les fenêtres de maintenance, les associations et les automatismes planifiés, la politique attachée à votre utilisateur, groupe ou rôle doit autoriser les actions suivantes :

- `ssm:DescribeMaintenanceWindows`
- `ssm:DescribeMaintenanceWindowExecution`
- `ssm:DescribeAutomationExecutions`
- `ssm:ListAssociations`

> **Note**
>
> Les entrées Change Calendar 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. Les fenêtres de maintenance, State Manager les associations et les automatisations ne sont pas partagées.

Utilisation des Change Calendar

Vous pouvez utiliser la console AWS Systems Manager pour ajouter, gérer ou supprimer des entrées dans Change Calendar, une fonctionnalité de AWS Systems Manager. Vous pouvez également importer des événements à partir de fournisseurs de calendrier tiers pris en charge en important un fichier iCalendar (`.ics`) exporté à partir du calendrier source. Vous pouvez utiliser l'opération d'API `GetCalendarState` ou la commande `get-calendar-state` AWS Command Line Interface (AWS CLI) pour obtenir des informations sur l'état de Change Calendar à un moment spécifique.

Rubriques

Création d'un calendrier de modifications

Lorsque vous créez une entrée dans Change Calendar, une fonctionnalité de AWS Systems Manager, vous créez un document Systems Manager (document SSM) au format `text`.

Pour créer un calendrier de modifications

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Change Calendar.

3. Sélectionnez Create calendar (Créer un calendrier).

   -ou-

   Si la page d'accueil Change Calendar s'ouvre en premier, sélectionnez Create change calendar (Créer un calendrier de modification).

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. Par exemple : **`support-off-hours`**. Vous ne pouvez pas mettre à jour ce nom après avoir créé l'entrée de calendrier.
5. (Facultatif) Pour Description, saisissez une description de votre entrée de calendrier.
6. (Facultatif) Dans la zone Import calendar (Importer un calendrier), sélectionnez Choose file (Choisir un fichier) pour sélectionner un fichier iCalendar (`.ics`) exporté à partir d'un fournisseur de calendriers tiers. L'importation du fichier ajoutera ses événements à votre calendrier.

   Les fournisseurs pris en charge comprennent Google Calendar, Microsoft Outlook et iCloud Calendar.

   Pour de plus amples informations, veuillez consulter Importation d'événements à partir de fournisseurs de calendrier tiers.
7. Dans Calendar type (Type de calendrier), sélectionnez 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.
8. (Facultatif) Dans Événements de gestion de modification, sélectionnez Ajouter au calendrier des événements de gestion de modification. Cette sélection affiche toutes les fenêtres de maintenance planifiées, les associations State Manager, les flux de travail d'automatisation et les demandes de modification Change Manager dans l'affichage de votre calendrier mensuel.

> **Tip**
>
> Si par la suite, vous souhaitez supprimer définitivement ces types d'événements de l'affichage du calendrier, modifiez le calendrier, décochez cette case, puis choisissez Enregistrer.

9. Sélectionnez Create calendar (Créer un calendrier).

   Une fois l'entrée de calendrier créée, Systems Manager affiche votre entrée de calendrier dans la liste Change Calendar. 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 créé ou importé au moins un événement. Pour plus d'informations sur la création d'un événement, consultez Création d'un événement Change Calendar. Pour plus d'informations sur l'importation d'évènements, consultez Importation d'événements à partir de fournisseurs de calendrier tiers.

Création et gestion d'événements dans Change Calendar

Après avoir créé un calendrier dans AWS Systems Manager Change Calendar, vous pouvez créer, mettre à jour et supprimer des événements inclus dans votre calendrier ouvert ou fermé. Change Calendar est une fonctionnalité de AWS Systems Manager.

> **Tip**
>
> Comme alternative à la création d'événements directement dans la console Systems Manager, vous pouvez importer un fichier iCalendar (`.ics`) à partir d'une application de calendrier tierce prise en charge. Pour plus d'informations, consultez Importation et gestion d'événements à partir de calendriers tiers.

Rubriques

- Création d'un événement Change Calendar
- Mise à jour d'un événement Change Calendar
- Suppression d'un événement Change Calendar

Création d'un événement Change Calendar

Lorsque vous ajoutez un événement à une entrée dans Change Calendar, une fonctionnalité de AWS Systems Manager, 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. (Vous pouvez également créer un événement consultatif, qui sert uniquement un rôle d'information dans le calendrier.)

Actuellement, vous ne pouvez créer un événement Change Calendar qu'à l'aide de la console. Les événements sont ajoutés au document Change Calendar que vous créez lorsque vous créez une entrée Change Calendar.

Pour créer un événement Change Calendar

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Change Calendar.
3. Dans la liste des calendriers, sélectionnez 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, sélectionnez 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. Par exemple : **`nighttime-hours`**.
6. Pour Description, saisissez une description de votre événement. Par exemple, **`The support team isn't available during these hours`**.
7. (Facultatif) Si vous souhaitez que cet événement ne serve que de notification visuelle ou de rappel, cochez la case Advisory (Recommandation). Les événements consultatifs ne jouent aucun rôle fonctionnel dans votre calendrier. Ils servent uniquement à des fins d'information pour ceux qui consultent votre calendrier.
8. Pour Event start date (Date de début de l'événement), saisissez ou sélectionnez 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).

9. Pour Event end date (Date de fin de l'événement), saisissez ou sélectionnez 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).
10. Pour Schedule time zone (Fuseau horaire de planification), sélectionnez 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).
11. (Facultatif) Pour créer un événement qui se répète chaque jour, chaque semaine ou chaque mois, activez l'option Recurrence (Récurrence), puis spécifiez la fréquence et la date de fin facultative de la récurrence.
12. Sélectionnez 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.

Mise à jour d'un événement Change Calendar

Procédez comme suit pour mettre à jour un événement Change Calendar dans la console AWS Systems Manager. Change Calendar est une fonctionnalité de AWS Systems Manager.

Pour mettre à jour un événement Change Calendar

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Change Calendar.
3. Dans la liste des calendriers, sélectionnez 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, sélectionnez Events (Événements).
5. Dans la page du calendrier, sélectionnez l'événement que vous souhaitez modifier.

> **Tip**
>
> 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 située en haut à droite.

6. Dans Event details (Détails de l'événement), sélectionnez Edit (Modifier).

   Pour modifier le nom et la description de l'événement, ajoutez ou remplacez les valeurs de texte actuelles.

7. Pour modifier l'option Event start date (Date de début de l'événement), sélectionnez la date de début actuelle, puis une nouvelle date dans le calendrier. Pour modifier l'heure de début, sélectionnez l'heure de début actuelle, puis une nouvelle heure dans la liste.

8. Pour modifier l'option Event end date (Date de fin de l'événement), sélectionnez la date actuelle, puis une nouvelle date de fin dans le calendrier. Pour modifier l'heure de fin, sélectionnez l'heure de fin actuelle, puis une nouvelle heure dans la liste.

9. Pour modifier la valeur du champ Schedule time zone (Fuseau horaire de planification), sélectionnez un fuseau horaire à appliquer 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. (Facultatif) Si vous souhaitez que cet événement ne serve que de notification visuelle ou de rappel, cochez la case Advisory (Recommandation). Les événements consultatifs ne jouent aucun rôle fonctionnel dans votre calendrier. Ils servent uniquement à des fins d'information pour ceux qui consultent votre calendrier.

11. Sélectionnez Enregistrer. Vos modifications s'affichent sous l'onglet Events (Événements) de la page des détails de l'entrée de calendrier. Sélectionnez l'événement que vous avez mis à jour pour afficher vos modifications.

Suppression d'un événement Change Calendar

Vous pouvez supprimer un événement à la fois dans Change Calendar, une fonctionnalité de AWS Systems Manager, via la AWS Management Console.

> **Tip**
>
> Si vous avez sélectionné Ajouter au calendrier des événements de gestion de modification lors de la création du calendrier, vous pouvez procéder comme suit :
>
> - Pour masquer temporairement un type d'événement de gestion des modifications dans l'affichage du calendrier, choisissez le X correspondant au type en haut de l'aperçu mensuel.

- Pour supprimer définitivement ces types de l'affichage du calendrier, modifiez le calendrier, décochez Ajouter au calendrier des événements de gestion de modification, puis choisissez Enregistrer. La suppression des types de l'affichage du calendrier ne les supprime pas de votre compte.

Pour supprimer un événement Change Calendar

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Change Calendar.
3. Dans la liste des calendriers, sélectionnez 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, sélectionnez Events (Événements).
5. Dans la page du calendrier, sélectionnez l'événement que vous souhaitez supprimer.

   > **Tip**
   >
   > 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 située en haut à droite.

6. Dans la page Event details (Détails de l'événement), sélectionnez Delete (Supprimer). Lorsque vous êtes invité à confirmer que vous souhaitez supprimer l'événement, sélectionnez Confirm (Confirmer).

Importation et gestion d'événements à partir de calendriers tiers

Comme alternative à la création d'événements directement dans la console AWS Systems Manager, vous pouvez importer un fichier iCalendar (`.ics`) à partir d'une application de calendrier tierce prise en charge. Votre calendrier peut inclure à la fois des événements importés et des événements créés par vos soins dans Change Calendar, une fonctionnalité de AWS Systems Manager.

Avant de commencer

Avant de tenter d'importer un fichier de calendrier, vérifiez les exigences et les contraintes suivantes :

Format de fichier de calendrier

Seuls les fichiers iCalendar (`.ics`) valides sont pris en charge.

Fournisseurs de calendriers pris en charge

Seuls les fichiers `.ics` exportés à partir des fournisseurs de calendriers tiers suivants sont pris en charge :

- Google Calendar (Instructions d'exportation)
- Microsoft Outlook (Instructions d'exportation)
- iCloud Calendar (Instructions d'exportation)

Taille de fichier

Vous pouvez importer n'importe quel nombre de fichiers `.ics` valides. Toutefois, la taille totale de tous les fichiers importés pour chaque calendrier ne peut pas dépasser 64 Ko.

> **Tip**
>
> Pour réduire la taille du fichier `.ics`, vérifiez que vous n'exportez que les détails de base de vos entrées de calendrier. Si nécessaire, réduisez la longueur de la période de temps exportée.

Fuseau horaire

Outre un nom de calendrier, un fournisseur de calendrier et au moins un événement, votre fichier `.ics` exporté doit indiquer le fuseau horaire du calendrier. Dans le cas contraire, ou si l'identification du fuseau horaire pose problème, vous serez invité à spécifier un fuseau horaire après l'importation du fichier.

Limitation d'événements récurrents

Le fichier `.ics` que vous exportez peut contenir des événements récurrents. Cependant, si une ou plusieurs occurrences d'un événement récurrent ont été supprimées dans le calendrier source, l'importation échouera.

Rubriques

- Importation d'événements à partir de fournisseurs de calendrier tiers

- Mise à jour de tous les événements d'un fournisseur de calendrier tiers
- Suppression de tous les événements importés d'un calendrier tiers

Importation d'événements à partir de fournisseurs de calendrier tiers

Procédez comme suit pour importer un fichier iCalendar (`.ics`) à partir d'une application de calendrier tierce prise en charge. Les événements contenus dans le fichier sont incorporés dans les règles de votre calendrier ouvert ou fermé. Vous pouvez importer un fichier dans le nouveau calendrier que vous créez avec Change Calendar (une fonctionnalité de AWS Systems Manager) ou dans un calendrier existant.

Après avoir importé le fichier `.ics`, vous pouvez en supprimer des événements individuels via l'interface Change Calendar. Pour plus d'informations, consultez Suppression d'un événement Change Calendar. Vous pouvez aussi supprimer tous les événements du calendrier source en supprimant le ficher `.ics`. Pour plus d'informations, consultez Suppression de tous les événements importés d'un calendrier tiers.

Pour importer des événements à partir de fournisseurs de calendrier tiers

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Change Calendar.
3. Pour démarrer avec un nouveau calendrier, sélectionnez Create calendar (Créer un calendrier). Dans la zone Import calendar (Importer un calendrier, sélectionnez Choose file (Choisir un fichier). Pour obtenir des informations sur les autres étapes de création d'un calendrier, consultez Création d'un calendrier de modifications.

   -ou-

   Pour importer des événements tiers dans un calendrier existant, sélectionnez le nom d'un calendrier existant pour l'ouvrir.
4. Sélectionnez Actions, Edit (Actions, modifier), puis dans la zone Import calendar (Importer le calendrier), sélectionnez Choose file (Choisir un fichier).
5. Sur votre ordinateur local, accédez au fichier `.ics` exporté et sélectionnez-le.
6. Si vous y êtes invité, pour Select a time zone (Sélectionner un fuseau horaire), sélectionnez le fuseau horaire qui s'applique au calendrier.
7. Choisissez Enregistrer.

Mise à jour de tous les événements d'un fournisseur de calendrier tiers

Si plusieurs événements sont ajoutés à votre calendrier source, ou supprimés de celui-ci, une fois son fichier iCalendar `.ics` importé, vous pouvez refléter ces modifications dans Change Calendar. Tout d'abord, réexportez le calendrier source, puis importez le nouveau fichier dans Change Calendar, une fonctionnalité de AWS Systems Manager. Les événements de votre calendrier de modifications seront mis à jour pour refléter le contenu du fichier le plus récent.

Pour mettre à jour tous les événements provenant d'un fournisseur de calendrier tiers

1. Dans votre calendrier tiers, ajoutez ou supprimez des événements de sorte qu'ils soient reflétés dans Change Calendar, puis réexportez le calendrier vers un nouveau fichier `.ics`.
2. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
3. Dans le panneau de navigation, sélectionnez Change Calendar.
4. Dans la liste des calendriers, sélectionnez le nom du calendrier.
5. Sélectionnez Choisir un fichier, puis recherchez et sélectionnez le fichier `.ics` de remplacement.
6. En réponse à la notification de l'écrasement du fichier existant, sélectionnez Confirm (Confirmer).

Suppression de tous les événements importés d'un calendrier tiers

Si vous ne voulez plus qu'aucun des événements importés à partir d'un fournisseur tiers figure dans votre calendrier, vous pouvez supprimer le fichier iCalendar `.ics` importé.

Pour supprimer tous les événements importés d'un calendrier tiers

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Change Calendar.
3. Dans la liste des calendriers, sélectionnez le nom du calendrier.
4. Dans la zone Import calendar (Importer un calendrier), sous My imported calendars (Mes calendriers importés), recherchez le nom du calendrier importé, puis sélectionnez le X dans sa carte.
5. Choisissez Enregistrer.

Mise à jour d'un calendrier de modifications

Vous pouvez mettre à jour la description d'un calendrier de modifications, mais pas son nom. Bien que vous puissiez modifier du statut 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 du statut Open by default (Ouvert par défaut) d'un calendrier par Closed by default (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 un calendrier de modifications, vous modifiez le document Change Calendar créé lors de la création de l'entrée. Change Calendar est une fonctionnalité de AWS Systems Manager.

Pour mettre à jour un calendrier de modifications

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Change Calendar.
3. Dans la liste des calendriers, sélectionnez le nom du calendrier à mettre à jour.
4. Sur la page des détails du calendrier, sélectionnez Actions, Edit (Actions, modifier).
5. Dans Description, vous pouvez modifier le texte de la description. Vous ne pouvez pas modifier le nom d'un calendrier de modifications.
6. Pour modifier du statut du calendrier, sélectionnez 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 au calendrier. Avant de modifier le type de calendrier, vous devez vérifier auprès des autres utilisateurs Change Calendar 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. Sélectionnez Enregistrer.

Votre 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 Création d'un événement Change Calendar.

Partage d'un calendrier de modifications

Vous pouvez partager un calendrier dansChange Calendar, une fonctionnalité de AWS Systems Manager, avec d'autres personnes à Comptes AWS l'aide de la AWS Systems Manager console. Lorsque vous partagez un calendrier, ce calendrier est en lecture seule pour les utilisateurs du compte partagé. Les fenêtres de maintenance, State Manager les associations et les automatisations ne sont pas partagées.

Pour partager un calendrier de modifications

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Change Calendar.
3. Dans la liste des calendriers, sélectionnez le nom du calendrier à partager.
4. Sur la page des détails du calendrier, sélectionnez l'onglet Sharing (Partage).
5. Sélectionnez Actions, Share (Actions, partager).
6. Dans Partager le calendrier, pour ID de compte, entrez le numéro d'identification d'un identifiant valide Compte AWS, puis choisissez Partager.

   Les utilisateurs du compte partagé peuvent lire le calendrier des modifications, mais ne peuvent y apporter aucune modification.

Suppression d'un calendrier de modifications

Vous pouvez supprimer un calendrier dans Change Calendar, une fonctionnalité de AWS Systems Manager, via la console Systems Manager ou la AWS Command Line Interface (AWS CLI). La suppression d'un calendrier des modifications supprime tous les événements associés.

Pour supprimer un calendrier de modifications

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.

2. Dans le panneau de navigation, sélectionnez Change Calendar.
3. Dans la liste des calendriers, sélectionnez le nom du calendrier à supprimer.
4. Dans la page des détails du calendrier, sélectionnez Actions, Delete (Actions, Supprimer). Lorsque vous êtes invité à confirmer que vous souhaitez supprimer le calendrier, sélectionnez Delete (Supprimer).

Obtenir le statut d'un calendrier de modifications

Vous pouvez obtenir le statut global d'un calendrier ou son statut à un moment donné dans Change Calendar, une fonctionnalité de AWS Systems Manager. Vous pouvez également afficher le prochain passage de l'état `OPEN` à l'état `CLOSED` (ou inversement) du calendrier.

Vous ne pouvez effectuer cette tâche qu'à l'aide de l'opération d'API `GetCalendarState`. La procédure décrite dans cette section utilise l'AWS Command Line Interface (AWS CLI).

Pour obtenir le statut d'un calendrier de modifications

- Exécutez la commande suivante pour afficher l'état d'une ou plusieurs calendriers à un moment spécifique. Le paramètre `--calendar-names` est obligatoire, mais `--at-time` est facultatif. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations.

  Linux & macOS

  ```
  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.

  ```
  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.

```
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 `OPEN` uniquement si tous les calendriers de la demande sont ouverts. Si un ou plusieurs calendriers de la demande sont fermés, le statut renvoyé est `CLOSED`.

Ajouter des dépendances Change Calendar à des runbooks Automation

Pour que les actions Automation respectent Change Calendar, une fonctionnalité de AWS Systems Manager, ajoutez une étape dans un runbook Automation qui utilise l'action `aws:assertAwsResourceProperty`. Configurez l'action pour exécuter `GetCalendarState` afin de vérifier qu'une entrée de calendrier spécifique est à l'état souhaité (`OPEN` ou `CLOSED`). Le runbook 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 runbook Automation qui ne peut pas passer à l'étape suivante, `LaunchInstance`, tant que l'état du calendrier ne correspond pas à `OPEN`, l'état spécifié dans `DesiredValues`.

Voici un exemple.

```
mainSteps:
  - name: MyCheckCalendarStateStep
    action: 'aws:assertAwsResourceProperty'
    inputs:
      Service: ssm
      Api: GetCalendarState
      CalendarNames: ["arn:aws:ssm:us-east-2:123456789012: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: python3.8
...
```

Résolution des problèmes de Change Calendar

Utilisez les informations suivantes pour essayer de résoudre les problèmes liés à Change Calendar, une des fonctionnalités de AWS Systems Manager.

Rubriques

- Erreur Échec d'importation du calendrier

Erreur Échec d'importation du calendrier

Problème : lors de l'importation d'un fichier iCalendar (`.ics`), le système signale que l'importation du calendrier a échoué.

- Solution 1 : vérifiez que vous importez bien un fichier exporté à partir d'un fournisseur de calendrier tiers pris en charge, notamment :
  - Google Calendar (Instructions d'exportation)
  - Microsoft Outlook (Instructions d'exportation)
  - iCloud Calendar (Instructions d'exportation)
- Solution 2 : si votre calendrier source contient des événements récurrents, vérifiez qu'aucune occurrence individuelle de l'événement n'a été annulée ou supprimée. Actuellement, Change Calendar ne prend pas en charge l'importation d'événements récurrents avec des annulations individuelles. Pour résoudre le problème, supprimez l'événement récurrent du calendrier source, réexportez le calendrier et réimportez-le dans Change Calendar, puis ajoutez l'événement récurrent via l'interface Change Calendar. Pour plus d'informations, consultez Création d'un événement Change Calendar.
- Solution 3 : vérifiez que votre calendrier source contient au moins un événement. Les téléchargements de fichiers `.ics` exempts d'événements échouent.
- Solution 4 : si le système signale que l'importation a échoué parce que le fichier `.ics` est trop volumineux, vérifiez que vous n'exportez que les détails de base de vos entrées de calendrier. Si nécessaire, réduisez la longueur de la période de temps que vous exportez.
- Solution 5 : si Change Calendar n'est pas en mesure de déterminer le fuseau horaire du calendrier exporté lorsque vous tentez de l'importer à partir de l'onglet Events (Événements), le message suivant peut s'afficher : « L'importation du calendrier a échoué. Change Calendar n'a pas pu localiser un fuseau horaire valide. Vous pouvez importer le calendrier à partir du menu Editi (Modifier). Dans ce cas, sélectionnez Actions, Edit (Actions, Modifier), puis essayez d'importer le fichier à partir de la page Edit calendar (Modifier le calendrier).
- Solution 6 : ne modifiez pas le fichier `.ics` avant importation. Les données du calendrier peuvent être corrompues si vous tentez de modifier le contenu du fichier. Si vous avez modifié le fichier avant de tenter de l'importer, exportez à nouveau le calendrier à partir du calendrier source, puis retentez le téléchargement.

AWS Systems Manager Maintenance Windows

Maintenance Windows, une fonctionnalité de AWS Systems Manager, vous aide à définir un calendrier indiquant quand effectuer des actions potentiellement perturbatrices sur vos nœuds, 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.

Vous pouvez ainsi planifier des actions sur de nombreux autres types de AWS ressources, tels que les buckets Amazon Simple Storage Service (Amazon S3), les AWS Key Management Service files d'attente Amazon Simple AWS KMS Queue Service (Amazon SQS), les clés (), etc. Maintenance Windows

Pour obtenir la liste complète des types de ressources pris en charge que vous pouvez inclure dans une cible de fenêtre de maintenance, consultez les sections Ressources que vous pouvez utiliser avec AWS Resource Groups et Éditeur de balises dans le guide de AWS Resource Groups l'utilisateur. Pour vos premiers pas dans Maintenance Windows, ouvrez Systems Manager console. Dans le panneau de navigation, sélectionnez Maintenance Windows.

> **Note**
>
> State Manager et Maintenance Windows peuvent effectuer certains types de mises à jour similaires sur vos nœuds gérés. Votre choix dépend de la nécessité d'automatiser la conformité du système ou d'effectuer des tâches hautement prioritaires et sensibles au temps pendant les périodes que vous spécifiez.
> Pour plus d'informations, consultez Choisir entre State Manager et Maintenance Windows.

Chaque fenêtre de maintenance comporte un calendrier, une durée maximale, un ensemble de cibles enregistrées (les nœuds gérés ou autres AWS ressources sur lesquelles on agit) et 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 sélectionner le fuseau horaire international pour 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 Options de planification de la fenêtre de maintenance et de période active.

Pour de plus amples informations sur l'utilisation de l'option `--schedule`, veuillez consulter Référence : Expressions Cron et Rate pour Systems Manager.

Types de tâches pris en charge

Avec les fenêtres de maintenance, vous pouvez exécuter quatre types de tâches :

- Commandes dans Run Command, une fonctionnalité de Systems Manager

  Pour plus d'informations sur Run Command, consultez AWS Systems Manager Run Command.
- Flux de travail dans Automation, une fonctionnalité de Systems Manager

  Pour de plus amples informations sur les flux de travail Automation, veuillez consulter AWS Systems Manager Automatisation.
- Fonctions dans AWS Lambda

  Pour plus d'informations sur les fonctions Lambda, consultez Mise en route avec Lambda dans le Guide du développeur AWS Lambda .
- Tâches dans AWS Step Functions

> **Note**
>
> Les tâches de la fenêtre de maintenance prennent uniquement en charge les flux de travail Step Functions Standard State Machine. Ils ne prennent pas en charge les flux de travail Express State Machine. Pour plus d'informations sur les types de flux de travail basés sur des machines à états, consultez la section Flux de travail standard et express dans le guide du AWS Step Functions développeur.

Pour plus d'informations sur Step Functions, consultez le Guide du développeur AWS Step Functions.

> **Note**
>
> Une ou plusieurs cibles doivent être spécifiées pour des tâches de fenêtre de maintenance de type Run Command. En fonction de la tâche, les cibles sont facultatives pour les autres types de tâches de la fenêtre de maintenance (Automation AWS Lambda, et AWS Step Functions).

Pour de plus amples informations sur l'exécution de tâches qui ne spécifient pas de cibles, consultez Enregistrement de tâches de fenêtre de maintenance sans cibles.

Cela signifie que vous pouvez utiliser les fenêtres de maintenance pour effectuer des tâches telles que les suivantes sur les cibles que vous avez sélectionnées.

- Installer ou mettre à jour des applications.
- Appliquer des correctifs.
- Installer ou mettre à jour l'SSM Agent.
- Exécutez PowerShell des commandes et des scripts shell Linux à l'aide d'une Run Command tâche Systems Manager.
- Créer des Amazon Machine Images (AMIs), amorcer des logiciels et configurer des nœuds à l'aide d'une tâche Systems Manager Automation.
- Exécutez AWS Lambda des fonctions qui appellent des actions supplémentaires, telles que l'analyse de vos nœuds à la recherche de mises à jour de correctifs.
- Exécutez des machines d' AWS Step Functions état pour effectuer des tâches telles que la suppression d'un nœud d'un environnement Elastic Load Balancing, l'application de correctifs au nœud, puis le réajout du nœud dans l'environnement Elastic Load Balancing.
- Ciblez les nœuds hors ligne en spécifiant un groupe de AWS ressources comme cible.

EventBridge soutien

Cette fonctionnalité de Systems Manager est prise en charge en tant que type d'événement dans les EventBridge règles d'Amazon. Pour plus d'informations, consultez Surveillance d'événements Systems Manager avec Amazon EventBridge et Référence : modèles et types d'événements Amazon EventBridge pour Systems Manager.

Table des matières

- Options de planification de la fenêtre de maintenance et de période active
- Enregistrement de tâches de fenêtre de maintenance sans cibles
- Résolution des problèmes liés aux fenêtres de maintenance

Configuration de Maintenance Windows

Avant que les utilisateurs de votre Compte AWS site puissent créer et planifier des tâches de la fenêtre de maintenance à l'aide Maintenance Windows d'une fonctionnalité de AWS Systems Manager, ils doivent disposer des autorisations nécessaires.

Avant de commencer

Pour effectuer les tâches de la section, vous avez besoin de l'une ou des deux ressources suivantes déjà configurées :

- Autorisations attribuées à une entité IAM (utilisateur, rôle ou groupe). Ces entités devraient déjà disposer d'autorisations générales pour travailler avec les fenêtres de maintenance. Cela peut être fait en attribuant la politique IAM `AmazonSSMFullAccess` aux utilisateurs ou groupes, ou une autre politique IAM fournissant un ensemble plus restreint d'autorisations d'accès pour Systems Manager qui couvre les tâches de la fenêtre de maintenance.
- (Facultatif) Pour les fenêtres de maintenance qui exécutent des tâches Run Command, vous pouvez choisir d'envoyer des notifications d'état Amazon Simple Notification Service (Amazon SNS). Run Command est une fonctionnalité de System Manager. Si vous souhaitez utiliser cette option, configurez la rubrique Amazon SNS avant de terminer ces tâches de configuration. 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 d'état du Systems Manager à l'aide des notifications Amazon SNS.

Présentation des tâches de configuration

Pour accorder les autorisations dont les utilisateurs ont besoin pour enregistrer des fenêtres de maintenance, un administrateur effectue les tâches suivantes. (Les instructions complètes sont fournies dans Utiliser la console pour configurer les autorisations pour les fenêtres de maintenance).

Tâche 1 : créer une politique à utiliser avec le rôle de fenêtre de maintenance personnalisée

Les tâches de la fenêtre de maintenance nécessitent un rôle IAM afin de fournir les autorisations requises pour s'exécuter sur les ressources cibles. Les types de tâches que vous exécutez et vos autres exigences opérationnelles déterminent le contenu de cette politique.

Nous fournissons une politique de base que vous pouvez adapter dans la rubrique Tâche 1 : création d'une politique pour votre fonction du service de fenêtre d'entretien personnalisée.

Tâche 2 : créer une fonction du service personnalisée pour les tâches de fenêtre de maintenance

La politique que vous créez dans la tâche 1 est attachée au rôle de la fenêtre de maintenance que vous créez dans la tâche 2. Lorsque les utilisateurs enregistrent une tâche de fenêtre de maintenance, ils spécifient cette fonction du service personnalisée dans le cadre de la configuration de la tâche. Les autorisations de ce rôle permettent à Systems Manager d'exécuter des tâches dans les fenêtres de maintenance en votre nom.

> **Important**
>
> Auparavant, la console Systems Manager vous permettait de choisir le rôle lié au service IAM AWS géré `AWSServiceRoleForAmazonSSM` à utiliser comme rôle de maintenance pour vos tâches. L'utilisation de ce rôle et de la politique associée, `AmazonSSMServiceRolePolicy`, pour les tâches de la fenêtre de maintenance n'est plus recommandée. Si vous utilisez ce rôle pour des tâches de fenêtre de maintenance maintenant, nous vous encourageons à cesser de l'utiliser. Au lieu de cela, créez votre propre rôle IAM permettant la communication entre Systems Manager et d'autres Services AWS lorsque les tâches de votre fenêtre de maintenance sont exécutées.

Tâche 3 : octroyer des autorisations d'utilisation de la fonction du service aux utilisateurs qui enregistrent les tâches de la fenêtre de maintenance

L'octroi aux utilisateurs d'autorisations pour accéder au rôle personnalisé de la fenêtre de maintenance leur permet de l'utiliser avec leurs tâches de fenêtre de maintenance. Cela vient s'ajouter aux autorisations que vous leur avez déjà accordées pour utiliser les commandes de l'API Systems Manager associées à Maintenance Windows cette fonctionnalité. Ce rôle indique les autorisations nécessaires pour exécuter une tâche de fenêtre de maintenance. Par conséquent, un utilisateur ne peut pas attribuer de tâches à une fenêtre de maintenance à l'aide de votre fonction du service personnalisée sans la possibilité de transmettre ces autorisations IAM.

Tâche 4 : (Facultatif) Rejeter explicitement les autorisations pour les utilisateurs qui ne sont pas autorisés à enregistrer les tâches de la fenêtre de maintenance

Vous pouvez refuser l'`ssm:RegisterTaskWithMaintenanceWindow`autorisation aux utilisateurs de votre site Compte AWS auxquels vous ne souhaitez pas enregistrer des tâches dans les fenêtres de maintenance. Cela fournit une couche de prévention supplémentaire pour les utilisateurs qui ne devraient pas enregistrer les tâches de la fenêtre de maintenance.

Rubriques

- Utiliser la console pour configurer les autorisations pour les fenêtres de maintenance

Utiliser la console pour configurer les autorisations pour les fenêtres de maintenance

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 : création d'une politique pour votre fonction du service de fenêtre d'entretien personnalisée
- Tâche 2 : création d'une fonction du service personnalisée pour les fenêtres de maintenance (console)
- Tâche 3 : configurer des autorisations pour les utilisateurs autorisés à enregistrer des tâches de fenêtre de maintenance (console)
- Tâche 4 : configurer des autorisations pour les utilisateurs qui ne sont pas autorisés à enregistrer des tâches de fenêtre de maintenance

Tâche 1 : création d'une politique pour votre fonction du service de fenêtre d'entretien personnalisée

Vous pouvez utiliser la politique suivante au format JSON pour créer la politique à utiliser avec votre rôle de fenêtre de maintenance. Vous attachez cette politique au rôle que vous créez ultérieurement dans Tâche 2 : création d'une fonction du service personnalisée pour les fenêtres de maintenance (console).

> **Important**
>
> En fonction des tâches et des types de tâches exécutées par vos fenêtres de maintenance, il se peut que vous n'ayez pas besoin de toutes les autorisations de cette politique, et que vous deviez inclure des autorisations supplémentaires.

Pour créer une politique pour votre fonction du service de fenêtre de maintenance personnalisée

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le volet de navigation, choisissez Policies, puis Create Policy.
3. Sélectionnez l'onglet JSON.
4. Remplacez le contenu par défaut par ce qui suit :

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:SendCommand",
                "ssm:CancelCommand",
                "ssm:ListCommands",
                "ssm:ListCommandInvocations",
                "ssm:GetCommandInvocation",
                "ssm:GetAutomationExecution",
                "ssm:StartAutomationExecution",
                "ssm:ListTagsForResource",
                "ssm:GetParameters"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "states:DescribeExecution",
                "states:StartExecution"
            ],
            "Resource": [
                "arn:aws:states:*:*:execution:*:*",
                "arn:aws:states:*:*:stateMachine:*"
```

```
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "lambda:InvokeFunction"
            ],
            "Resource": [
                "arn:aws:lambda:*:*:function:*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "resource-groups:ListGroups",
                "resource-groups:ListGroupResources"
            ],
            "Resource": [
                "*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "tag:GetResources"
            ],
            "Resource": [
                "*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": [
                        "ssm.amazonaws.com"
                    ]
                }
            }
        }
    ]
```

```
}
```

5. Modifiez le contenu JSON en fonction des besoins des tâches de maintenance que vous exécutez dans votre compte. Les modifications que vous apportez sont spécifiques à vos opérations planifiées.

   Par exemple :

   - Vous pouvez fournir des noms de ressources Amazon Resource Name (ARN) pour des fonctions et des machines d'état spécifiques au lieu d'utiliser des qualificatifs génériques (*).
   - Si vous ne prévoyez pas d'exécuter des tâches AWS Step Functions, vous pouvez supprimer les autorisations `states` et (ARN).
   - Si vous ne prévoyez pas d'exécuter des tâches AWS Lambda, vous pouvez supprimer les autorisations `lambda` et les ARN.
   - Si vous ne prévoyez pas d'exécuter des tâches d'automatisation, vous pouvez supprimer les autorisations `ssm:GetAutomationExecution` et `ssm:StartAutomationExecution`.
   - Ajoutez des autorisations supplémentaires qui peuvent être nécessaires à l'exécution des tâches. Par exemple, certaines actions Automation utilisent des piles AWS CloudFormation. Par conséquent, les autorisations `cloudformation:CreateStack`, `cloudformation:DescribeStacks` et `cloudformation:DeleteStack` sont requises.

     Autre exemple, le runbook d'automatisation `AWS-CopySnapshot` requiert des autorisations pour créer un instantané Amazon Elastic Block Store (Amazon EBS). Par conséquent, la fonction du service a besoin de l'autorisation `ec2:CreateSnapshot`.

     Pour plus d'informations sur les autorisations de rôle requises par les runbooks d'automatisation, consultez les descriptions du runbook dans la référence du runbook d'automatisation d'AWS Systems Manager.

6. Une fois les révisions de la politique terminées, choisissez Next : Tags (Suivant : balises).
7. (Facultatif) Ajoutez une ou plusieurs paires clé-valeur de balise afin d'organiser, de suivre ou de contrôler l'accès pour ce rôle, puis sélectionnez Next : Review (Suivant : Vérifier).
8. Pour Name (Nom), saisissez un nom qui identifie cette politique comme étant la politique que la fonction du service Maintenance Windows que vous créez des utilisations. Par exemple : **`my-maintenance-window-role-policy`**.
9. Choisissez Create policy (Créer une politique), et notez le nom que vous avez spécifié pour la politique. Vous y faites référence dans la procédure suivante, Tâche 2 : création d'une fonction du service personnalisée pour les fenêtres de maintenance (console).

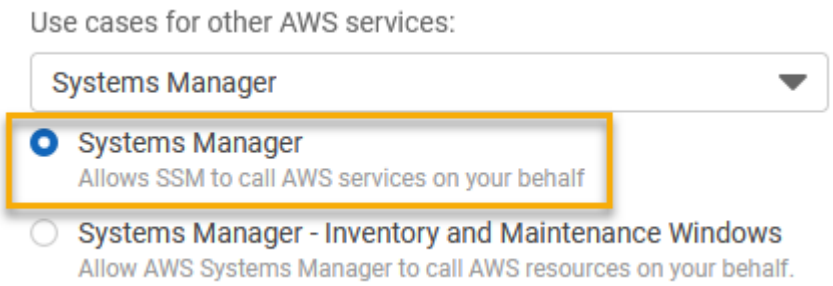
Tâche 2 : création d'une fonction du service personnalisée pour les fenêtres de maintenance (console)

Utilisez la procédure suivante pour créer un rôle de service personnalisé pour les Maintenance Windows, afin que Systems Manager puisse exécuter des tâches Maintenance Windows en votre nom. Vous attacherez la politique que vous avez créée dans la tâche précédente au rôle de service que vous créez.

> **Important**
>
> Auparavant, la console Systems Manager vous offrait la possibilité de choisir le rôle `AWSServiceRoleForAmazonSSM` lié au service IAM géré par AWS à utiliser comme rôle de maintenance pour vos tâches. L'utilisation de ce rôle et de la politique associée, `AmazonSSMServiceRolePolicy`, pour les tâches de la fenêtre de maintenance n'est plus recommandée. Si vous utilisez ce rôle pour des tâches de fenêtre de maintenance maintenant, nous vous encourageons à cesser de l'utiliser. Au lieu de cela, créez votre propre rôle IAM permettant la communication entre Systems Manager et d'autres Services AWS lorsque les tâches de votre fenêtre de maintenance sont exécutées.

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, sélectionnez Rôles, puis Créer un rôle.
3. Pour Select trusted entity (Sélectionner une entité de confiance), effectuez les choix suivants :
   1. Pour Type d'entité de confiance, choisissez Service AWS
   2. Pour Cas d'utilisation d'autres services AWS, choisissez Systems Manager
   3. Choisissez Systems Manager, comme illustré dans l'image suivante.



4. Choisissez Next (Suivant).

5. Dans la zone de recherche, saisissez le nom de la politique que vous avez créée dans Tâche 1 : création d'une politique pour votre fonction du service de fenêtre d'entretien personnalisée, cochez la case située à côté de son nom, puis choisissez Next (Suivant).
6. Dans Role name (Nom du rôle), saisissez un nom qui identifie celui-ci en tant que rôle Maintenance Windows. Par exemple : **`my-maintenance-window-role`**.
7. (Facultatif) Modifiez la description du rôle par défaut pour refléter l'objectif de ce rôle. Par exemple : **`Performs maintenance window tasks on your behalf`**.
8. (Facultatif) Ajoutez une ou plusieurs paires clé-valeur de balise afin d'organiser, de suivre ou de contrôler l'accès pour ce rôle, puis sélectionnez Next: Review (Suivant : Vérifier).
9. Sélectionnez Créer un rôle. Le système vous renvoie à la page Rôles.
10. Sélectionnez le nom du rôle que vous venez de créer.
11. Cliquez sur l'onglet Trust relationships (Relations d'approbation) puis vérifiez que la politique suivante s'affiche dans la boîte Trusted entities (Entités de confiance).

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

12. Copier le nom du rôle et notez le nom de rôle et la valeur de l'ARN de la zone Summary (Récapitulatif). Les utilisateurs de votre compte spécifient ces informations lorsqu'ils créent des fenêtres de maintenance.

Tâche 3 : 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 au service Systems Manager pour exécuter les opérations de tâche réelles. C'est le rôle que le service endossera lorsqu'il exécutera des tâches en votre nom.

Avant cela, pour enregistrer la tâche elle-même, affectez la politique PassRole IAM à une entité IAM (comme un compte ou un groupe). Cela permet à l'entité IAM (utilisateur ou groupe) de spécifier, lors de l'enregistrement de ces tâches dans la fenêtre de maintenance, le rôle à utiliser lors de l'exécution des tâches. Pour en savoir plus, reportez-vous à Octroi d'autorisations à un utilisateur pour transférer un rôle à un Service AWS dans le Guide de l'utilisateur IAM.

Configurer des autorisations pour les utilisateurs autorisés à enregistrer des tâches de fenêtre de maintenance

Si une entité IAM (utilisateur, rôle ou groupe) est configurée avec des autorisations d'administrateur, l'utilisateur ou le rôle a accès aux fenêtres de maintenance. Pour les entités IAM sans autorisations d'administrateur, un administrateur doit accorder les autorisations suivantes à l'entité IAM. Voici les autorisations minimales requises pour enregistrer des tâches dans une fenêtre de maintenance :

- La politique gérée `AmazonSSMFullAccess`, ou une politique qui fournit des autorisations comparables.
- Les autorisations `iam:PassRole` et `iam:ListRoles` suivantes.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::account-id:role/my-maintenance-window-role"
        },
        {
            "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/"
        }
    ]
}
```

my-maintenance-window-role représente le nom du rôle de fenêtre de maintenance personnalisé que vous avez créé précédemment.

account-id représente 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.

Pour activer l'accès, ajoutez des autorisations à vos utilisateurs, groupes ou rôles :

- Utilisateurs et groupes dans AWS IAM Identity Center :

  Créez un jeu d'autorisations. Suivez les instructions de la rubrique Création d'un jeu d'autorisations du Guide de l'utilisateur AWS IAM Identity Center.

- Utilisateurs gérés dans IAM par un fournisseur d'identité :

  Créez un rôle pour la fédération d'identité. Pour plus d'informations, voir la rubrique Création d'un rôle pour un fournisseur d'identité tiers (fédération) du Guide de l'utilisateur IAM.

- Utilisateurs IAM :
  - Créez un rôle que votre utilisateur peut assumer. Suivez les instructions de la rubrique Création d'un rôle pour un utilisateur IAM du Guide de l'utilisateur IAM.
  - (Non recommandé) Attachez une politique directement à un utilisateur ou ajoutez un utilisateur à un groupe d'utilisateurs. Suivez les instructions de la rubrique Ajout d'autorisations à un utilisateur (console) du Guide de l'utilisateur IAM.

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, sélectionnez User groups (Groupes d'utilisateurs).
3. Dans la liste de groupes, sélectionnez le nom du groupe auquel accorder l'autorisation `iam:PassRole`.
4. Sous l'onglet Permissions (Autorisations), sélectionnez Add permissions, Create inline policy (Ajouter des autorisations, créer une politique en ligne), puis l'onglet JSON.

5. Remplacez le contenu par défaut de la zone par ce qui suit.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::account-id:role/my-maintenance-window-role"
        },
        {
            "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/"
        }
    ]
}
```

my-maintenance-window-role représente le nom du rôle de fenêtre de maintenance personnalisé que vous avez créé précédemment.

account-id représente 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. Sélectionnez Examiner une politique.
7. Sur la page Review policy (Examiner une politique), saisissez un nom dans la zone Name (Nom) pour identifier la politique `PassRole`, tel que **`my-group-iam-passrole-policy`**, puis sélectionnez Create policy (Créer une politique).

Tâche 4 : configurer des autorisations pour les utilisateurs qui ne sont pas autorisés à enregistrer des tâches de fenêtre de maintenance

Selon que vous refusez l'autorisation `ssm:RegisterTaskWithMaintenanceWindow` à un utilisateur individuel ou à un groupe, utilisez l'une des procédures suivantes pour empêcher les utilisateurs d'enregistrer des tâches avec une fenêtre de maintenance.

Configurer des autorisations pour les utilisateurs qui ne sont pas autorisés à enregistrer des tâches de fenêtre de maintenance

- Un administrateur doit ajouter les restrictions suivantes à l'entité IAM.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": "ssm:RegisterTaskWithMaintenanceWindow",
            "Resource": "*"
        }
    ]
}
```

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, sélectionnez User groups (Groupes d'utilisateurs).
3. Dans la liste de groupes, sélectionnez le nom du groupe auquel refuser l'autorisation `ssm:RegisterTaskWithMaintenanceWindow`.
4. Sous l'onglet Permissions (Autorisations), sélectionnez Add permissions, create inline policy (Ajouter des autorisations, Créer une politique en ligne).
5. Cliquez sur l'onglet JSON, puis remplacez le contenu par défaut de la case par ce qui suit.

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

```
            "Action": "ssm:RegisterTaskWithMaintenanceWindow",
            "Resource": "*"
        }
    ]
}
```

6. Sélectionnez Examiner une politique.
7. Sur la page Review policy (Examiner une politique), saisissez un nom dans la zone Name (Nom) pour identifier cette politique, tel que **`my-groups-deny-mw-tasks-policy`**, puis sélectionnez Create policy (Créer une politique).

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 Maintenance Windows, une des fonctionnalités de AWS Systems Manager. Pour de plus amples informations, veuillez consulter Configuration de Maintenance Windows.

Rubriques

- Créer une fenêtre de maintenance (console)
- Affecter des cibles à une fenêtre de maintenance (console)
- Attribuer des tâches à une fenêtre de maintenance (console)
- Désactivation ou activation d'une fenêtre de maintenance
- Mise à jour ou suppression de ressources de fenêtre de maintenance (console)

Créer une fenêtre de maintenance (console)

Dans cette procédure, vous créez une fenêtre de maintenance dans Maintenance Windows, une fonctionnalité de AWS Systems Manager. Vous pouvez spécifier 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 Options de planification de la fenêtre de maintenance et de période active.
> Pour de plus amples informations sur l'utilisation de l'option `--schedule`, veuillez consulter Référence : Expressions Cron et Rate pour Systems Manager.

Créer une fenêtre de maintenance (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Maintenance Windows.
3. Sélectionnez Create maintenance window (Créer une fenêtre de maintenance).
4. Pour Name (Nom), saisissez un nom évocateur pour vous aider à identifier cette fenêtre de maintenance.
5. (Facultatif) Pour Description, saisissez une description pour définir la façon dont cette fenêtre de maintenance sera utilisée.
6. (Facultatif) Si vous souhaitez autoriser l'exécution d'une tâche de fenêtre de maintenance sur des nœuds gérés, même si vous n'avez pas enregistré ces nœuds comme cibles, sélectionnez Allow unregistered targets (Autoriser les cibles non enregistrées).

   Lorsque vous sélectionnez cette option, vous pouvez sélectionner les nœuds non enregistrés (par ID de nœud) lorsque vous enregistrez une tâche auprès de la fenêtre de maintenance.

   Si vous ne sélectionnez 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.

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 Window start date (Fenêtre de date de début), 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.

    > **Note**
    >
    > Vous ne pouvez pas spécifier une date et une heure de début antérieures.

11. (Facultatif) Pour Window start date (Fenêtre de date de début), 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) Dans Schedule timezone (Fuseau horaire de planification), indiquez le fuseau horaire sur lequel doit se baser l'exécution des fenêtres de maintenance planifiées, au format IANA (Internet Assigned Numbers Authority). Par 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) Pour Schedule offset (Décalage de planification), saisissez le nombre de jours à attendre après la date et l'heure spécifiées par une expression cron ou rate avant d'exécuter la fenêtre de maintenance. Vous pouvez spécifier une valeur entre un et six jours.

    > **Note**
    >
    > Cette option n'est disponible que si vous avez spécifié une planification en saisissant manuellement une expression cron ou rate.

14. (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`

15. Sélectionnez 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.

> **Note**
>
> Si une seule tâche de fenêtre de maintenance est enregistrée avec plusieurs cibles, ses appels de tâches se produisent en séquence et non en parallèle. Si votre tâche doit

s'exécuter sur plusieurs cibles simultanément, enregistrez une tâche pour chaque cible individuellement et attribuez à chaque tâche le même niveau de priorité.

Affecter des cibles à une fenêtre de maintenance (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Maintenance Windows.
3. Dans la liste des fenêtres de maintenance, sélectionnez la fenêtre de maintenance à laquelle ajouter les cibles.
4. Sélectionnez 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 les informations sur le propriétaire, spécifiez les informations à inclure dans tout EventBridge événement Amazon déclenché lors de l'exécution de tâches pour ces cibles dans cette fenêtre de maintenance.

   Pour plus d'informations sur l'utilisation EventBridge pour surveiller les événements de Systems Manager, consultezSurveillance d'événements Systems Manager avec Amazon EventBridge.
8. Dans la zone Targets (Cibles), sélectionnez l'une des options décrites dans le tableau suivant.

Option	Description
Spécification de balises d'instance	Dans les zones Specify instance tags (Spécifier les balises d'instance), spécifiez une ou plusieurs clés de balise et (facultat if) des valeurs de balise qui ont été ou seront ajoutées aux nœuds gérés de votre compte. Lorsqu'elle s'exécute, la fenêtre de maintenance tente d'exécuter des tâches sur tous les nœuds gérés auxquels ces balises ont été ajoutées. Si vous spécifiez plusieurs clés de balise, un nœud doit être balisé avec toutes les clés et

Option	Description
	valeurs de balise que vous décidez d'inclure dans le groupe cible.
Choix manuel des instances	Dans la liste, cochez la case située en regard de chaque nœud que vous souhaitez inclure dans la cible de fenêtre de maintenance. La liste inclut tous les nœuds de votre compte qui sont configurés pour être utilisés avec Systems Manager. Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés pour obtenir des conseils de dépannage. Pour les appareils de périphérie, les serveurs sur site et les machines virtuelles, consultez Utilisation de Systems Manager dans des environnements hybrides et multicloud.

Option	Description
Pour choisir un groupe de ressources	Pour Resource group (Groupe de ressources), sélectionnez 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 : • Que sont les groupes de ressources ? dans le Guide de l'utilisateur AWS Resource Groups • Groupes de ressources et balisage pour AWS dans l'AWS News Blog. Pour Resource types (Types de ressource), sélectionnez jusqu'à cinq types de ressources disponibles, ou l'option 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`

Option	Description
	Mais plus tard, lorsque vous ajoutez des tâches à la fenêtre de maintenance, vous incluez uniquement des tâches qui exécutent des actions sur les nœuds, comme l'application d'un référentiel de correctifs ou le redémarrage d'un nœud. Dans le journal de fenêtre de maintenance, une erreur peut signaler qu'aucun compartiment Amazon Simple Storage Service (Amazon S3) ou table Amazon DynamoDB n'a été trouvé. Toutefois, la fenêtre de maintenance continue d'exécuter des tâches sur les nœuds de votre groupe de ressources.

9. Sélectionnez Register target (Enregistrer la cible).

Si vous souhaitez attribuer d'autres cibles à cette fenêtre de maintenance, sélectionnez l'onglet Targets (Cibles), puis sélectionnez Register targets (Enregistrer des cibles). Avec cette option, vous pouvez choisir d'autres méthodes de ciblage. Par exemple, si vous cibliez auparavant les nœuds par ID de nœud, vous pouvez enregistrer de nouvelles cibles et de nouveaux nœuds cibles en spécifiant les balises appliquées aux nœuds gérés ou en choisissant des types de ressources 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 qui sont effectuées lors de l'exécution d'une fenêtre de maintenance.

Les quatre types de tâches peuvent être ajoutés à une fenêtre de maintenance :

- Commandes de l'AWS Systems Manager Run Command
- Flux de travail Systems Manager Automation
- AWS Step Functions tâches
- AWS Lambda fonctions

> **Important**
>
> La politique IAM pour Maintenance Windows exige que vous donniez un préfixe `SSM` à la fonction Lambda (ou alias). Avant de procéder à l'enregistrement de ce type de tâche, mettez à jour son nom AWS Lambda pour inclure`SSM`. Par exemple, si votre nom de fonction Lambda est `MyLambdaFunction`, remplacez-le par `SSMMyLambdaFunction`.

Pour attribuer des tâches à une fenêtre de maintenance

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Maintenance Windows.
3. Dans la liste des fenêtres de maintenance, sélectionnez une fenêtre de maintenance.
4. Sélectionnez Actions, puis sélectionnez 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 Automation
   - Enregistrer une tâche Lambda
   - Enregistrer la tâche Step Functions

   > **Note**
   >
   > Les tâches de la fenêtre de maintenance prennent uniquement en charge les flux de travail Step Functions Standard State Machine. Ils ne prennent pas en charge les flux de travail Express State Machine. Pour plus d'informations sur les types de flux de travail basés sur des machines à états, consultez la section Flux de travail standard et express dans le guide du AWS Step Functions développeur.

5. (Facultatif) Pour Name (Nom), saisissez un nom pour la tâche.
6. (Facultatif) Sous Description, entrez une description.
7. Dans New task invocation cutoff (Nouvelle limite d'appel de tâches), si vous ne souhaitez pas que de nouveaux appels de tâches soient lancés une fois le délai attribué à la fenêtre de maintenance écoulé, sélectionnez Enabled (Activé).

Lorsque cette option n'est pas activée, la tâche continue de s'exécuter une fois le délai écoulé et lance de nouveaux appels de tâches jusqu'à ce que celles-ci soient accomplies.

> **Note**
>
> L'état des tâches qui ne sont pas terminées lorsque vous activez cette option est `TIMED_OUT`.

8. Pour cette étape, suivez les sous-étapes correspondant au type de tâche sélectionné.

   Fonctionnalité Exécuter la commande

   1. Dans la liste des documents de commande, choisissez le document de commande de Systems Manager (document SSM) qui définit les tâches à exécuter.
   2. Pour Version du document, sélectionnez la version de document à utiliser.
   3. 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.

   Automation

   1. Dans la liste des documents d'automatisation, choisissez le manuel d'automatisation qui définit les tâches à exécuter.
   2. Pour Document version (Version du document), sélectionnez la version du runbook à utiliser.
   3. 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.

   Lambda

   1. Dans la zone Paramètres Lambda, choisissez une fonction Lambda dans la liste.
   2. (Facultatif) Indiquez un contenu pour Payload (Charge utile), Client Context (Contexte client) ou Qualifier (Qualificateur) que vous souhaitez inclure.

> **Note**
>
> Dans certains cas, vous pouvez utiliser un pseudo paramètre dans votre `Payload` valeur. Ensuite, lorsque la tâche de fenêtre de maintenance s'exécute, elle transmet les valeurs correctes au lieu des espaces réservés aux pseudo-paramètres. Pour plus d'informations, veuillez consulter Utilisation de pseudo-paramètres lors de l'enregistrement des tâches de la fenêtre de maintenance.

3. 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.

Step Functions

1. Dans la zone des paramètres Step Functions, choisissez une machine à états dans la liste.
2. (Facultatif) Indiquez un nom pour l'exécution de la machine d'état et tout contenu pour Input (Entrée) que vous souhaitez inclure.

> **Note**
>
> Dans certains cas, vous pouvez utiliser un pseudo paramètre dans votre `Input` valeur. Ensuite, lorsque la tâche de fenêtre de maintenance s'exécute, elle transmet les valeurs correctes au lieu des espaces réservés aux pseudo-paramètres. Pour plus d'informations, veuillez consulter Utilisation de pseudo-paramètres lors de l'enregistrement des tâches de la fenêtre de maintenance.

3. 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.

9. Dans la zone Cibles, sélectionnez l'une des options suivantes :

- Sélection de groupes cibles enregistrés : sélectionnez une ou plusieurs cibles de fenêtre de maintenance que vous avez enregistrées avec la fenêtre de maintenance actuelle.
- Sélection de cibles non enregistrées : sélectionnez les ressources disponibles une par une comme cibles pour la tâche.

Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés pour obtenir des conseils de dépannage.

- Cible de tâche non requise : les cibles de la tâche peuvent déjà être spécifiées dans d'autres fonctions pour toutes les tâches à l'exception des tâches de type Run Command.

  Spécifiez une ou plusieurs cibles pour des tâches de fenêtre de maintenance de type Run Command. En fonction de la tâche, les cibles sont facultatives pour les autres types de tâches de la fenêtre de maintenance (Automation AWS Lambda, et AWS Step Functions). Pour de plus amples informations sur l'exécution de tâches qui ne spécifient pas de cibles, consultez Enregistrement de tâches de fenêtre de maintenance sans cibles.

  > **Note**
  >
  > Dans la plupart des cas, il est inutile de spécifier explicitement une cible pour une tâche d'automatisation. Par exemple, supposons que vous créiez une tâche de type Automation pour mettre à jour une Amazon Machine Image (AMI) pour Linux à l'aide du runbook `AWS-UpdateLinuxAmi`. Lorsque la tâche s'exécute, l'AMI est mise à jour avec les derniers packages de distribution Linux et les logiciels Amazon disponibles. Ces mises à jour sont déjà installées sur les nouvelles instances créées à partir de l'AMI. Comme l'ID de l'AMI à mettre à jour est spécifié dans les paramètres d'entrée du runbook, il est inutile de spécifier à nouveau une cible dans la tâche de la fenêtre de maintenance.

10. Tâches d'automatisation uniquement :

    Dans Paramètres d'entrée, fournissez des valeurs pour tous les paramètres requis ou facultatifs nécessaires à l'exécution de votre tâche.

    > **Note**
    >
    > Dans certains cas, vous pouvez utiliser un pseudo-paramètre pour certaines valeurs de paramètres d'entrée. Ensuite, lorsque la tâche de fenêtre de maintenance s'exécute, elle transmet les valeurs correctes au lieu des espaces réservés aux pseudo-paramètres. Pour plus d'informations, veuillez consulter Utilisation de pseudo-paramètres lors de l'enregistrement des tâches de la fenêtre de maintenance.

11. Pour Rate control (Contrôle de débit) :

    - Dans Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage de nœuds gérés sur lesquels exécuter simultanément la commande.

      > **Note**
      >
      > Si vous avez sélectionné des cibles en spécifiant les balises appliquées aux nœuds gérés ou en spécifiant des groupes de ressources AWS , et que vous n'êtes pas certain du nombre de nœuds gérés ciblés, limitez le nombre de cibles autorisées à exécuter simultanément le document en indiquant un pourcentage.

    - Dans Error threshold (Seuil d'erreur), indiquez quand arrêter l'exécution de la commande sur les autres nœuds gérés après l'échec de celle-ci sur un certain nombre ou un certain pourcentage de nœuds. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les nœuds gérés sur lesquels la commande est toujours en cours de traitement peuvent également envoyer des erreurs.

12. (Facultatif) Pour le rôle de service IAM, choisissez un rôle qui fournira des autorisations à Systems Manager lors de l'exécution d'une tâche pendant la fenêtre de maintenance.

    Si vous ne spécifiez aucun ARN de rôle de service, Systems Manager utilise un rôle lié à un service dans votre compte. S'il n'existe aucun rôle lié à un service approprié pour Systems Manager dans votre compte, il est créé lorsque la tâche est enregistrée avec succès.

    > **Note**
    >
    > Pour améliorer le niveau de sécurité, nous vous recommandons vivement de créer une politique personnalisée et un rôle de service personnalisé pour exécuter les tâches de votre fenêtre de maintenance. La politique peut être conçue pour fournir uniquement les autorisations nécessaires pour les tâches spécifiques de votre fenêtre de maintenance. Pour plus d'informations, consultez Utiliser la console pour configurer les autorisations pour les fenêtres de maintenance.

13. Run Commandtâches uniquement :

    (Facultatif) Pour Output options (Options de sortie), procédez de l'une des manières suivantes :

    - Cochez la case Activer l'écriture dans S3 pour enregistrer la sortie de la commande dans un fichier. Saisissez les noms de compartiment et de préfixe (dossier) dans les zones.

- Cochez la case CloudWatch de sortie pour écrire la sortie complète dans Amazon CloudWatch Logs. Entrez le nom d'un groupe de CloudWatch journaux Logs.

> **Note**
>
> Les autorisations qui permettent d'écrire des données dans un compartiment S3 ou dans CloudWatch des journaux sont celles du profil d'instance attribué au nœud, et non celles de l'utilisateur IAM effectuant cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager. En outre, si le compartiment ou le groupe de journaux S3 spécifié se trouve dans un autre compartiment Compte AWS, vérifiez que le profil d'instance associé au nœud dispose des autorisations nécessaires pour écrire dans ce compartiment.

14. Run Commandtâches uniquement :

    Dans la section SNS notifications (Notifications SNS), si vous souhaitez envoyer des notifications sur le statut d'exécution des commandes, cochez la case Enable SNS notifications (Activer les notifications SNS).

    Pour plus d'informations sur la configuration des notifications Amazon SNS pour Run Command, consultez Surveillance des changements d'état du Systems Manager à l'aide des notifications Amazon SNS.

15. Run Commandtâches uniquement :

    Dans la section Parameters (Paramètres), spécifiez les paramètres du document.

> **Note**
>
> Dans certains cas, vous pouvez utiliser un pseudo-paramètre pour certaines valeurs de paramètres d'entrée. Ensuite, lorsque la tâche de fenêtre de maintenance s'exécute, elle transmet les valeurs correctes au lieu des espaces réservés aux pseudo-paramètres. Pour plus d'informations, veuillez consulter Utilisation de pseudo-paramètres lors de l'enregistrement des tâches de la fenêtre de maintenance.

16. Run Commandet tâches d'automatisation uniquement :

    (Facultatif) Dans la zone CloudWatch d'alarme, pour Nom de l'alarme, choisissez une CloudWatch alarme existante à appliquer à votre tâche de surveillance.

Si l'alarme est activée, la tâche est arrêtée.

> **Note**
>
> Pour associer une CloudWatch alarme à votre tâche, le principal IAM qui exécute la tâche doit être autorisé à effectuer l'`iam:createServiceLinkedRole`action. Pour plus d'informations sur les CloudWatch alarmes, consultez la section Utilisation des CloudWatch alarmes Amazon.

17. En fonction de votre type de tâche, choisissez l'une des options suivantes :

    - Enregistrer une tâche d'exécution de commande
    - Enregistrer la tâche Automation
    - Enregistrer une tâche Lambda
    - Enregistrer la tâche Step Functions

Désactivation ou activation d'une fenêtre de maintenance

Vous pouvez désactiver ou activer une fenêtre de maintenance dans Maintenance Windows, une des fonctionnalités de AWS Systems Manager. Vous pouvez choisir une fenêtre de maintenance à la fois pour désactiver ou activer son exécution. Vous pouvez également sélectionner plusieurs ou toutes les fenêtres de maintenance pour les activer et les désactiver.

Cette section explique comment désactiver ou activer une fenêtre de maintenance à l'aide de la console Systems Manager. Pour des exemples illustrant comment procéder à l'aide de AWS Command Line Interface (AWS CLI), voirDidacticiel : Mettre à jour une fenêtre de maintenance (AWS CLI).

Rubriques

- Désactivation d'une fenêtre de maintenance (console)
- Activation d'une fenêtre de maintenance (console)

Désactivation d'une fenêtre de maintenance (console)

Vous pouvez désactiver une fenêtre de maintenance pour suspendre une tâche pendant une période spécifiée, et elle restera disponible pour être réactivée ultérieurement.

Pour désactiver une fenêtre de maintenance

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Maintenance Windows.
3. À l'aide de la case en regard de la fenêtre de maintenance à désactiver, sélectionnez une ou plusieurs fenêtres de maintenance.
4. Choisissez Désactiver la fenêtre de maintenance dans le menu Actions. Le système vous invite à confirmer vos actions.

Activation d'une fenêtre de maintenance (console)

Vous pouvez activer une fenêtre de maintenance pour reprendre une tâche.

> **Note**
>
> Si la fenêtre de maintenance utilise un barème tarifaire et que la date de début est actuellement définie sur une date et une heure passées, la date et l'heure actuelles sont utilisées comme date de début de la fenêtre de maintenance. Vous pouvez modifier la date de début de la fenêtre de maintenance avant ou après son activation. Pour plus d'informations, veuillez consulter Mise à jour ou suppression de ressources de fenêtre de maintenance (console).

Pour activer une fenêtre de maintenance

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Maintenance Windows.
3. Cochez la case à côté de la fenêtre de maintenance pour l'activer.
4. Choisissez Actions, puis Activer la fenêtre de maintenance. Le système vous invite à confirmer vos actions.

Mise à jour ou suppression de ressources de fenêtre de maintenance (console)

Vous pouvez mettre à jour ou supprimer une fenêtre de maintenance dans Maintenance Windows, une des fonctionnalités de AWS Systems Manager. 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 Systems Manager. Pour voir des exemples de la façon de procéder à l'aide de l'AWS Command Line Interface (AWS CLI), consultez Didacticiel : Mettre à jour une fenêtre de maintenance (AWS CLI).

Rubriques

- Mise à jour ou suppression d'une fenêtre de maintenance (console)
- Mise à jour de fenêtres de maintenance cibles ou annulation de leur enregistrement (console)
- Mise à jour de tâches de fenêtre de maintenance ou annulation de leur enregistrement (console)

Mise à jour ou suppression d'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, sélectionnez Maintenance Windows.
3. Sélectionnez le bouton en regard de la fenêtre de maintenance à mettre à jour ou supprimer, puis exécutez l'une des tâches suivantes :

   - Choisissez Supprimer. Le système vous invite à confirmer vos actions.
   - Sélectionnez Edit (Modifier). Sur la page Edit maintenance window (Modifier la fenêtre de maintenance), modifiez les valeurs et options souhaitées, puis sélectionnez Save changes (Enregistrer les modifications).

     Pour de plus amples informations sur les choix de configuration que vous pouvez effectuer, veuillez consulter Créer une fenêtre de maintenance (console).

Mise à jour de fenêtres de maintenance cibles ou annulation de leur enregistrement (console)

Vous pouvez mettre à jour les cibles d'une fenêtre de maintenance ou annuler leur enregistrement. 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, sélectionnez Maintenance Windows.
3. Sélectionnez le nom de la fenêtre de maintenance à mettre à jour, puis l'onglet Tabs (Onglets), et exécutez l'une des tâches suivantes :

   - Pour mettre à jour les cibles, sélectionnez le bouton en regard de la cible à mettre à jour, puis Edit (Modifier).
   - Pour annuler l'enregistrement de cibles, cliquez sur le bouton à côté de la cible dont l'enregistrement doit être annulé, puis sélectionnez Deregister target (Désenregistrer la cible). Dans la boite de dialogue Désenregistrer une cible de fenêtre de maintenance, sélectionnez Annuler l'enregistrement.

Mise à jour de tâches de fenêtre de maintenance ou annulation de leur enregistrement (console)

Vous pouvez mettre à jour les tâches d'une fenêtre de maintenance ou annuler leur enregistrement. 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 Automation et Run Command, 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 Run Command.

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, sélectionnez Maintenance Windows.
3. Sélectionnez le nom de la fenêtre de maintenance que vous souhaitez mettre à jour.
4. Sélectionnez l'onglet Tasks (Tâches), puis le bouton en regard de la tâche à mettre à jour.

5. Effectuez l'une des actions suivantes :

   - Pour annuler l'enregistrement d'une tâche, sélectionnez Deregister task (Annuler l'enregistrement de la tâche).
   - Pour modifier la tâche, sélectionnez Edit (Modifier). Modifiez les valeurs et options de votre choix, puis sélectionnez Edit task (Modifier la tâche).

Didacticiels Systems Manager Maintenance Windows (AWS CLI)

Cette section inclut des didacticiels qui vous aident à apprendre à utiliser le 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 :

- Configurer le AWS CLI sur votre machine locale — Avant de pouvoir exécuter des AWS CLI commandes, vous devez installer et configurer la CLI sur votre machine locale. Pour plus d'informations, consultez la section Installation ou mise à jour de la version la plus récente de l' AWS CLI et Installation d' AWS Tools for PowerShell.
- Vérifiez les rôles et les autorisations des fenêtres 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 la CLI. Pour plus d'informations, consultez Configuration de Maintenance Windows.
- Créer ou configurer une instance compatible avec Systems Manager – Pour suivre les didacticiels, vous avez besoin d'au moins une instance Amazon Elastic Compute Cloud (Amazon EC2), configurée pour une utilisation avec Systems Manager. Cela signifie que SSM Agent est installé sur l'instance et qu'un profil d'instance IAM pour Systems Manager est attaché à l'instance.

Nous recommandons de lancer une instance à partir d'une instance AWS gérée Amazon Machine Image (AMI) avec l'agent préinstallé. Pour plus d'informations, consultez Rechercher AMIs avec le SSM Agent préinstallé.

Pour plus d'informations sur l'installation de SSM Agent sur une instance, consultez les rubriques suivantes :

- Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Windows Server
- Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Linux

Pour plus d'informations sur la configuration des autorisations IAM pour Systems Manager sur votre instance, voir Configurer les autorisations d'instance requises pour Systems Manager.

- Créer des ressources supplémentaires en fonction des besoins – Run Command, une des fonctionnalités de Systems Manager, inclut nombreuses 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 Run Command simple que vous pourrez utiliser lors de votre première utilisation des didacticiels. Comme décrit précédemment dans cette rubrique, vous avez également besoin d'une instance (EC2) configurée pour une utilisation avec Systems Manager. Une fois que vous avez configuré cette instance, vous pouvez enregistrer une simple tâche Run Command.

  La fonctionnalité Systems Manager Maintenance Windows prend en charge les quatre types de tâches suivants :

  - Commandes de l'Run Command
  - Flux de travail Systems Manager Automation
  - AWS Lambda fonctions
  - AWS Step Functions tâches

  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 AWS Lambda fonction, créez la fonction Lambda avant de commencer ; pour une Run Command tâche, créez le compartiment S3 dans lequel vous pouvez enregistrer la sortie de commande (si vous prévoyez de le faire) ; etc.

Conserver une trace des ID de ressources

Au fur et à mesure que vous effectuez les tâches de ce AWS CLI didacticiel, suivez les identifiants de ressources générés 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 le format suivant :

```
{
   "WindowId":"mw-0c50858d01EXAMPLE"
}
```

Prenez note des ID suivants générés par le système, étant donné que 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)
- Didacticiel : Afficher des informations concernant les fenêtres de maintenance (AWS CLI)
- Didacticiel : Afficher les informations sur les tâches et les exécutions de tâches (AWS CLI)
- Didacticiel : Mettre à jour une fenêtre de maintenance (AWS CLI)
- Didacticiel : Supprimer une fenêtre de maintenance (AWS CLI)

Didacticiel : Créer et configurer une fenêtre de maintenance (AWS CLI)

Ce didacticiel montre comment utiliser l'AWS Command Line Interface (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 en *rouge* et en italique par vos propres options et identifiants. Par exemple, remplacez l'ID de la fenêtre de maintenance *mw-0c50858d01EXAMPLE* et l'ID de l'instance *i-02573cafcfEXAMPLE* par les ID des ressources que vous créez.

Table des matières

Étape 1 : Création de 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 Options de planification de la fenêtre de maintenance et de période active.
> Pour de plus amples informations sur l'utilisation de l'option `--schedule`, veuillez consulter Référence : Expressions Cron et Rate pour Systems Manager.

Pour créer une fenêtre de maintenance (AWS CLI)

1. Ouvrez l'AWS Command Line Interface (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 le démarrage de nouvelles tâches dans l'heure qui suit la fin de l'opération de 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 & macOS

```
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 retourne des informations telles que les suivantes.

```
{
   "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 retourne des informations telles que les suivantes.

```
{
   "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 : Enregistrement d'un nœud cible auprès de la fenêtre de maintenance (AWS CLI).

Étape 2 : Enregistrement d'un nœud cible auprès de 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 le nœud à mettre à jour pendant l'exécution de la fenêtre de maintenance.

Pour obtenir un exemple d'enregistrement de plusieurs nœuds à la fois à l'aide d'ID de nœud, des exemples d'utilisation de balises permettant d'identifier plusieurs nœuds ainsi que des exemples de spécification de groupes de ressources comme cibles, consultez Exemples : Enregistrement de cibles avec une fenêtre de maintenance.

> **Note**
>
> Vous devez déjà avoir créé une instance Amazon Elastic Compute Cloud (Amazon EC2) à utiliser dans cette étape, comme décrit dans les Prérequis du didacticiel Maintenance Windows.

Pour enregistrer un nœud cible auprès d'une fenêtre de maintenance (AWS CLI)

1. Exécutez la commande suivante sur votre machine locale. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   Linux & macOS

   ```
   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 retourne des informations telles que les suivantes.

   ```
   {
      "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 & macOS

   ```
   aws ssm describe-maintenance-window-targets \
       --window-id "mw-0c50858d01EXAMPLE"
   ```

   Windows

   ```
   aws ssm describe-maintenance-window-targets ^
       --window-id "mw-0c50858d01EXAMPLE"
   ```

   Le système retourne des informations telles que les suivantes.

```
{
    "Targets": [
        {
            "WindowId": "mw-0c50858d01EXAMPLE",
            "WindowTargetId": "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE",
            "ResourceType": "INSTANCE",
            "Targets": [
                {
                    "Key": "InstanceIds",
                    "Values": [
                        "i-02573cafcfEXAMPLE"
                    ]
                }
            ]
        }
    ]
}
```

Passez au Étape 3 : Enregistrement d'une tâche avec la fenêtre de maintenance (AWS CLI).

Exemples : Enregistrement de cibles avec une fenêtre de maintenance

Vous pouvez enregistrer un nœud individuel comme cible à l'aide de son ID de nœud, comme démontré dans Étape 2 : Enregistrement d'un nœud cible auprès de la fenêtre de maintenance (AWS CLI). Vous pouvez également enregistrer un ou plusieurs nœuds comme cibles à l'aide des commandes présentées sur cette page.

En général, deux méthodes permettent d'identifier les nœuds à utiliser comme cibles de fenêtre de maintenance : spécifier des nœuds individuels 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 nœuds et beaucoup d'autres types de ressources 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.

> **Note**
>
> Si une seule tâche de fenêtre de maintenance est enregistrée avec plusieurs cibles, ses appels de tâches se produisent en séquence et non en parallèle. Si votre tâche doit

s'exécuter sur plusieurs cibles simultanément, enregistrez une tâche pour chaque cible individuellement et attribuez à chaque tâche le même niveau de priorité.

Pour plus d'informations sur la création et la gestion des groupes de ressources, consultez Que sont les groupes de ressources ? dans le Guide de l'utilisateur AWS Resource Groups et Groupes de ressources et balisage pour AWS dans AWS News Blog.

Pour obtenir des informations sur les quotas pour Maintenance Windows, une fonctionnalité d'AWS Systems Manager, en plus de ceux spécifiés dans les exemples suivants, veuillez consulter la rubrique Quotas de service Systems Manager dans la Référence générale d'Amazon Web Services.

Exemple 1 : Enregistrement de plusieurs cibles à l'aide des ID de nœud

Exécutez la commande suivante sur votre machine locale pour enregistrer plusieurs nœuds comme cibles à l'aide de leurs ID de nœud. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

Linux & macOS

```
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 : particulièrement utile lorsque vous enregistrez pour la première fois un groupe unique de nœuds auprès d'une fenêtre de maintenance et qu'ils ne partagent pas de balise de nœud commune.

Quotas : vous pouvez spécifier un maximum de 50 nœuds pour chaque cible de fenêtre de maintenance.

Exemple 2 : Enregistrement de cibles à l'aide des balises de ressources appliquées aux nœuds

Exécutez la commande suivante sur votre machine locale pour enregistrer les nœuds qui sont déjà balisés avec une paire clé-valeur que vous avez attribuée. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

Linux & macOS

```
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 : particulièrement utile lorsque vous enregistrez pour la première fois un groupe unique de nœuds auprès d'une fenêtre de maintenance et qu'ils partagent une balise de nœud commune.

Quotas : vous pouvez spécifier jusqu'à cinq paires clé-valeur au total pour chaque cible. Si vous spécifiez plusieurs paires clé-valeur, un nœud doit être balisé avec toutes les clés et valeurs de balise que vous décidez d'inclure dans le groupe cible.

> **Note**
>
> Vous pouvez baliser un groupe de noeuds avec la clé de balise : `Patch GroupouPatchGroup` et attribuer les noeuds d'une valeur de clé commune, comme `my-patch-group`. (Utilisez `PatchGroup`, sans espace, si vous avez autorisé les balises dans les métadonnées d'instance EC2.) Patch Manager, une fonctionnalité de Systems Manager, évalue la clé `Patch Group` ou `PatchGroup` sur les nœuds afin d'aider à déterminer le référentiel de correctifs applicable à ces nœuds. Si votre tâche exécutera le document SSM `AWS-RunPatchBaseline` (ou le document SSM `AWS-ApplyPatchBaseline` hérité), spécifiez la même paire clé-valeur `Patch Group` ou `PatchGroup` lors de l'enregistrement des cibles avec une fenêtre de maintenance. Par exemple : `--target`

"Key=tag:PatchGroup,Values=*my-patch-group*. Cela vous permet d'utiliser une fenêtre de maintenance pour mettre à jour les correctifs d'un groupe de nœuds qui sont déjà associés au même référentiel de correctifs. Pour de plus amples informations, veuillez consulter À propos des groupes de correctifs.

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 nœuds qui disposent tous d'une ou plusieurs clés de balise, indépendamment de leurs valeurs de clé. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

Linux & macOS

```
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 nœuds 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, un nœud doit être balisé avec toutes les clés de balise que vous décidez d'inclure dans le groupe cible.

Exemple 4 : Enregistrement de 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. Remplacez *mw-0c50858d01EXAMPLE* avec vos propres informations. 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 & macOS

```
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 ressource seront ciblés par une fenêtre de maintenance, ou lorsque vous savez que le groupe de ressources contient uniquement les types de ressource sur lesquelles vos tâches exécutent des actions.

Quotas : Vous pouvez spécifier un seul groupe de ressources comme cible.

Exemple 5 : Enregistrement de cibles en filtrant des types de ressource 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. Remplacez *mw-0c50858d01EXAMPLE* avec vos propres informations. 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 & macOS

```
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 si 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 ressource et que vous voulez éviter les rapports d'erreur inutiles dans les journaux de votre fenêtre de maintenance.

Quotas : Vous pouvez spécifier un seul groupe de ressources comme cible.

Étape 3 : Enregistrement d'une tâche avec la fenêtre de maintenance (AWS CLI)

Dans cette étape du didacticiel, vous enregistrez une tâche AWS Systems Manager Run Command qui exécute la commande `df` sur votre instance Amazon Elastic Compute Cloud (Amazon 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 Amazon 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, le masque de sous-réseau et la passerelle par défaut pour les adaptateurs sur l'instance cible.

Lorsque vous êtes prêt à enregistrer d'autres types de tâches ou à utiliser plusieurs des options Systems Manager Run Command disponibles, consultez Exemples : Enregistrement de tâches avec une fenêtre de maintenance. 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. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations. 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 & macOS

   ```
   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 & macOS

```
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/
MyMaintenanceWindowServiceRole",
            "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 : Création de la fenêtre de maintenance (AWS CLI). 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 & macOS

```
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. Remplacez *mw-0c50858d01EXAMPLE* avec vos propres informations.

Linux & macOS

```
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 et Options de planification de la fenêtre de maintenance et de période active.
Pour plus d'informations sur l'utilisation de l'AWS Command Line Interface (AWS CLI) pour modifier une fenêtre de maintenance, consultez Didacticiel : Mettre à jour une fenêtre de maintenance (AWS CLI).

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).

À propos de la sortie de commande du didacticiel

L'utilisation de l'AWS CLI pour afficher la sortie de la commande Run Command 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 console Systems Manager ou dans un fichier journal stocké dans un compartiment Amazon Simple Storage Service [Amazon S3], si vous avez configuré la fenêtre de maintenance afin qu'elle stocke la sortie de commande à cet endroit.) Vous constaterez que la sortie de la commande df 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 : Enregistrement de tâches avec une fenêtre de maintenance

Vous pouvez enregistrer une tâche dansRun Command, une fonctionnalité de AWS Systems Manager, avec une fenêtre de maintenance à l'aide du AWS Command Line Interface (AWS CLI), comme illustré dans Enregistrer des tâches avec la fenêtre de maintenance. Vous pouvez également enregistrer des tâches pour les flux de travail, les AWS Lambda fonctions et les AWS Step Functions tâches de Systems Manager Automation, comme expliqué plus loin dans cette rubrique.

> **Note**
>
> Spécifiez une ou plusieurs cibles pour des tâches de fenêtre de maintenance de type Run Command. En fonction de la tâche, les cibles sont facultatives pour les autres types de tâches de la fenêtre de maintenance (Automation AWS Lambda, et AWS Step Functions). Pour de plus amples informations sur l'exécution de tâches qui ne spécifient pas de cibles, consultez Enregistrement de tâches de fenêtre de maintenance sans cibles.

Dans cette rubrique, nous donnons des exemples d'utilisation de la commande AWS Command Line Interface (AWS CLI) `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'cli-input-json option --

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 le nom de fichier par *`MyFile.json.`*

Linux & macOS

```
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, vous pouvez utiliser `{{TARGET_ID}}` et `{{RESOURCE_ID}}` pour transmettre des ID de ressources AWS à des tâches Automation, Lambda et Step Functions. Pour plus d'informations sur les pseudo-paramètres dans les contenus `--task-invocation-parameters`, consultez Utilisation de pseudo-paramètres lors de l'enregistrement des tâches de la fenêtre de maintenance.

Plus d'informations

- À propos des register-task-with-maintenance options -windows.
- register-task-with-maintenance-window dans la référence de commande de l'AWS CLI
- RegisterTaskWithMaintenanceWindow dans la Référence d'API AWS Systems Manager

Exemples d'enregistrement de tâches

Les sections suivantes fournissent un exemple de AWS CLI commande pour enregistrer un type de tâche pris en charge et un exemple JSON pouvant être utilisé avec l'`--cli-input-json`option.

Enregistrer une tâche Systems Manager Run Command

Les exemples suivants montrent comment enregistrer des tâches Systems Manager Run Command avec une fenêtre de maintenance à l'aide de l' AWS CLI.

Linux & macOS

```
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 l'option de fichier **`--cli-input-json`** :

```
{
    "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:region:123456789012:my-sns-topic-name",
                "NotificationEvents": [
```

```
                    "All"
                ],
                "NotificationType": "Invocation"
            },
            "OutputS3BucketName": "DOC-EXAMPLE-BUCKET",
            "OutputS3KeyPrefix": "S3-PREFIX",
            "TimeoutSeconds": 3600
        }
    }
}
```

Enregistrer une tâche Systems Manager Automation

Les exemples suivants montrent comment enregistrer des tâches Automation Systems Manager avec une fenêtre de maintenance à l'aide de l' AWS CLI :

AWS CLI commande :

Linux & macOS

```
aws ssm register-task-with-maintenance-window \
    --window-id "mw-0c50858d01EXAMPLE" \
    --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 --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" ^
    --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 --name "My-Restart-EC2-Instances-Automation-Task" ^
    --description "Automation task to restart EC2 instances"
```

Contenu JSON à utiliser avec l'option de fichier **`--cli-input-json`** :

```
{
    "WindowId": "mw-0c50858d01EXAMPLE",
        "TaskArn": "AWS-PatchInstanceWithRollback",
    "TaskType": "AUTOMATION","TaskInvocationParameters": {
        "Automation": {
            "DocumentVersion": "1",
            "Parameters": {
                "instanceId": [
                    "{{RESOURCE_ID}}"
                ]
            }
        }
    }
}
```

Enregistrer une tâche AWS Lambda

Les exemples suivants montrent comment enregistrer des tâches de fonctions Lambda avec une fenêtre de maintenance à l'aide de la 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 politique IAM pour Maintenance Windows exige que vous donniez un préfixe `SSM` à la fonction Lambda (ou alias). Avant de procéder à l'enregistrement de ce type de tâche, mettez à jour son nom AWS Lambda pour inclure`SSM`. Par exemple, si votre nom de fonction Lambda est `MyLambdaFunction`, remplacez-le par `SSMMyLambdaFunction`.

AWS CLI commande :

Linux & macOS

> **Important**
>
> Si vous utilisez la version 2 du AWS CLI, vous devez inclure l'option `--cli-binary-format raw-in-base64-out` dans la commande suivante si votre charge utile Lambda

n'est pas codée en base64. L'option `cli_binary_format` n'est disponible que dans la version 2. Pour plus d'informations à ce sujet et sur d'autres paramètres de AWS CLI `config` fichier, consultez la section Paramètres de `config` fichiers pris en charge dans le Guide de AWS Command Line Interface l'utilisateur.

```
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:region:123456789012:function:serverlessrepo-
SSMrestart-my-instances-C4JF9EXAMPLE" \
    --task-invocation-parameters '{"Lambda":{"Payload":"{\"InstanceId\":
\"{{RESOURCE_ID}}\",\"targetType\":\"{{TARGET_TYPE}}\"}","Qualifier": "$LATEST"}}'
```

PowerShell

Important

Si vous utilisez la version 2 du AWS CLI, vous devez inclure l'option `--cli-binary-format raw-in-base64-out` dans la commande suivante si votre charge utile Lambda n'est pas codée en base64. L'option `cli_binary_format` n'est disponible que dans la version 2. Pour plus d'informations à ce sujet et sur d'autres paramètres de AWS CLI `config` fichier, consultez la section Paramètres de `config` fichiers pris en charge dans le Guide de AWS Command Line Interface l'utilisateur.

```
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:region: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 de fichier **`--cli-input-json`** :

```
{
    "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 Step Functions

Les exemples suivants montrent comment enregistrer des tâches de machine d'état Step Functions avec une fenêtre de maintenance à l'aide de la AWS CLI.

> **Note**
>
> Les tâches de la fenêtre de maintenance prennent uniquement en charge les flux de travail Step Functions Standard State Machine. Ils ne prennent pas en charge les flux de travail Express State Machine. Pour plus d'informations sur les types de flux de travail basés sur des machines à états, consultez la section Flux de travail standard et express dans le guide du AWS Step Functions développeur.

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 politique AWS Identity and Access Management (IAM) pour Maintenance Windows exige que vous préfixiez les noms des machines d'état Step Functions par. `SSM` Avant de procéder à l'enregistrement de ce type de tâche, vous devez mettre à jour son nom AWS Step Functions pour l'inclure`SSM`. Par exemple, si le nom de votre machine d'état est `MyStateMachine`, remplacez-le par `SSMMyStateMachine`.

AWS CLI commande :

Linux & macOS

```
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:region: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"
```

PowerShell

```
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:region: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 de fichier **`--cli-input-json`** :

```
{
    "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 register-task-with-maintenance options -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 les autres options de commande, consultez register-task-with-maintenance-window dans la Référence Command AWS CLI .

A propos de l'option **`--task-arn`**

L'option `--task-arn` est utilisée pour spécifier la ressource sur laquelle la tâche opère. La valeur que vous spécifiez dépend du type de tâche que vous enregistrez, comme décrit dans le tableau suivant.

TaskArn formats pour les tâches de la fenêtre de maintenance

Type de tâche de fenêtre de maintenance	TaskArn valeur
RUN_COMMAND et **AUTOMATION**	`TaskArn` correspond au nom ou à l'Amazon Resource Name (ARN) du nom du document SSM. Par exemple : `AWS-RunBatchShellScript` -ou- `arn:aws:ssm:` *`region`*`:11112222 3333:document/My-Document` .
LAMBDA	`TaskArn` est le nom de fonction ou l'ARN. Par exemple : `SSMMy-Lambda-Function` -ou- `arn:aws:lambda:` *`region`*`:11112222 3333:function:SSMMyLambdaFu nction` . **Important** La politique IAM pour Maintenance Windows exige que vous donniez un préfixe `SSM` à la fonction Lambda (ou alias). Avant de procéder à l'enregistrement de ce type de tâche, mettez à jour son nom AWS Lambda pour inclure`SSM`. Par exemple, si votre nom de fonction Lambda est

Type de tâche de fenêtre de maintenance	TaskArn valeur
	`MyLambdaFunction` , remplacez-le par `SSMMyLambdaFunction` .
`STEP_FUNCTIONS`	`TaskArn` est l'ARN de la machine d'état. Par exemple : `arn:aws:states:us-east-2:111122223333:stateMachine:SSMMyStateMachine` . **Important** La politique IAM pour les fenêtres de maintenance exige que vous donniez un préfixe aux noms de machines d'état Step Functions avec `SSM`. Avant d'enregistrer ce type de tâche, vous devez mettre à jour son nom AWS Step Functions pour l'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`**

Le rôle AWS Systems Manager à assumer lors de l'exécution de la tâche de fenêtre de maintenance.

Pour de plus amples informations, veuillez consulter Configuration de Maintenance Windows

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 Utilisation de pseudo-paramètres lors de l'enregistrement des tâches de la fenêtre de maintenance.

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	Comment DocumentHash DocumentHashType NotificationConfig Sorties 3 BucketName OutPutS3 KeyPrefix Paramètres ServiceRoleArn TimeoutSeconds	`"TaskInvocationParameters": { "RunCommand": { "Comment": "My Run Command task comment", "DocumentHash": "6554ed3d--truncated--5EXAMPLE", "DocumentHashType": "Sha256", "NotificationConfig": { "NotificationArn": "arn:aws:sns: region:123456789012:my-sns-topic-name", "NotificationEvents": ["FAILURE"], "NotificationType": "Invocation" },`

Type de tâche de fenêtre de maintenance	Paramètres disponibles	Exemple
		"OutputS3BucketName": "DOC-EXAMPLE-BUCKET", "OutputS3KeyPrefix": " *S3-PREFIX* ", "Parameters": { "commands": ["Get-ChildItem$env:temp-Recurse|Remove-Item-Recurse-force"] }, "ServiceRoleArn": "arn:aws:iam::123456789012:role/MyMaintenanceWindowServiceRole", "TimeoutSeconds": 3600 } }

Type de tâche de fenêtre de maintenance	Paramètres disponibles	Exemple
Automatisation	DocumentVersion Paramètres	<pre>"TaskInvocationParameters": { "Automation": { "DocumentVersion": "3", "Parameters": { "instanceid": ["{{TARGET_ID}}"] } } }</pre>
LAMBDA	ClientContext Charge utile Qualificateur	<pre>"TaskInvocationParameters": { "Lambda": { "ClientContext": "ew0KICAi--truncated--0KIEXAMPLE", "Payload": "{ \"targetId\": \"{{TARGET_ID}}\", \"targetType\": \"{{TARGET_TYPE}}\" }", "Qualifier": "$LATEST" } }</pre>

Type de tâche de fenêtre de maintenance	Paramètres disponibles	Exemple
STEP_FUNCTIONS	Entrée Nom	`"TaskInvocationParameters": { "StepFunctions": { "Input": "{ \"targetId\": \"{{TARGET_ID}}\" }", "Name": "{{INVOCATION_ID}}" } }`

Didacticiel : Afficher des informations concernant 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 en *rouge* et en italique par vos propres options et identifiants. Par exemple, remplacez l'ID de la fenêtre de maintenance *mw-0c50858d01EXAMPLE* et l'ID de l'instance *i-02573cafcfEXAMPLE* par les ID des ressources que vous créez.

Pour obtenir des informations sur la configuration de l'AWS Command Line Interface (AWS CLI), consultez Installation, mise à jour et désinstallation d'AWS CLI et Configuration d'AWS CLI.

Exemples de commandes

- Exemples pour « describe-maintenance-windows »
- Exemples pour « describe-maintenance-window-targets »
- Exemples pour « describe-maintenance-window-tasks »
- Exemples pour « describe-maintenance-windows-for-target »
- Exemples pour « describe-maintenance-window-executions »

- Exemples pour « describe-maintenance-window-schedule »

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 retourne des informations telles que les suivantes.

```
{
   "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 retourne des informations telles que les suivantes.

```
{
```

```
   "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 retourne des informations telles que les suivantes.

```
{
    "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 retourne des informations telles que les suivantes.

```
{
    "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 & macOS

```
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 retourne des informations telles que les suivantes.

```
{
    "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 le document de commande SSM **`AWS-RunPowerShellScript`**

Exécutez la commande suivante.

Linux & macOS

```
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 retourne des informations telles que les suivantes.

```
{
   "Tasks":[
      {
         "ServiceRoleArn": "arn:aws:iam::111122223333:role/
MyMaintenanceWindowServiceRole",
         "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/
MyMaintenanceWindowServiceRole",
         "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 & macOS

```
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 retourne des informations telles que les suivantes.

```
{
   "Tasks":[
      {
         "ServiceRoleArn":"arn:aws:iam::111122223333:role/
MyMaintenanceWindowServiceRole",
         "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 Run Command

Exécutez la commande suivante.

Linux & macOS

```
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 retourne des informations telles que les suivantes.


```
{
    "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/
MyMaintenanceWindowServiceRole",
            "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/
MyMaintenanceWindowServiceRole",
            "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 relatives aux cibles ou tâches de fenêtre de maintenance associées à un nœud spécifique

Exécutez la commande suivante.

Linux & macOS

```
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 retourne des informations telles que les suivantes.

```
{
    "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 & macOS

```
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 retourne des informations telles que les suivantes.

```
{
    "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 & macOS

```
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 retourne des informations telles que les suivantes.

```
{
    "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 un nœud particulier

Exécutez la commande suivante.

Linux & macOS

```
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 retourne des informations telles que les suivantes.

```
{
    "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 le calendrier des fenêtres de maintenance pour les nœuds balisés avec une paire clé-valeur donnée

Exécutez la commande suivante.

Linux & macOS

```
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 retourne des informations telles que les suivantes.

```
{
    "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 & macOS

```
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 retourne des informations telles que les suivantes.

```
{
    "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 Command Line Interface (AWS CLI) pour afficher des informations détaillées sur les tâches de fenêtres de maintenance.

Si vous passez directement à ce didacticiel à partir de Didacticiel : Créer et configurer une fenêtre de maintenance (AWS CLI), 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 en *rouge* et en italique par vos propres options et identifiants. Par exemple, remplacez l'ID de la fenêtre de maintenance *mw-0c50858d01EXAMPLE* et l'ID de l'instance *i-02573cafcfEXAMPLE* par les ID des ressources 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 & macOS

   ```
   aws ssm describe-maintenance-window-executions \
       --window-id "mw-0c50858d01EXAMPLE"
   ```

Windows

```
aws ssm describe-maintenance-window-executions ^
    --window-id "mw-0c50858d01EXAMPLE"
```

Le système retourne des informations telles que les suivantes.

```
{
    "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 une exécution de tâche d'une fenêtre de maintenance.

   Linux & macOS

   ```
   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 retourne des informations telles que les suivantes.

```
{
    "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 afficher les tâches exécutées dans le cadre de l'exécution d'une fenêtre de maintenance.

Linux & macOS

```
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 retourne des informations telles que les suivantes.

```
{
    "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 & macOS

   ```
   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 retourne des informations telles que les suivantes.

   ```
   {
       "WindowExecutionId": "14bea65d-5ccc-462d-a2f3-e99c8EXAMPLE",
       "TaskExecutionId": "c9b05aba-197f-4d8d-be34-e73fbEXAMPLE",
       "TaskArn": "AWS-RunShellScript",
       "ServiceRole": "arn:aws:iam::111122223333:role/MyMaintenanceWindowServiceRole",
       "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 & macOS

   ```
   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 retourne des informations telles que les suivantes.

   ```
   {
       "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 le AWS Command Line Interface (AWS CLI) pour mettre à jour une fenêtre de maintenance. Il vous montre également comment mettre à jour différents types de tâches, y compris celles pour AWS Systems Manager Run Command et Automation 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 Mise à jour ou suppression de ressources de fenêtre de maintenance (console).

Tout au long des étapes de ce didacticiel, remplacez les valeurs en *rouge* et en italique par vos propres options et identifiants. Par exemple, remplacez l'ID de la fenêtre de maintenance *mw-0c50858d01EXAMPLE* et l'ID de l'instance *i-02573cafcfEXAMPLE* par les ID des ressources que vous créez.

Pour mettre à jour une fenêtre de maintenance (AWS CLI)

1. Ouvrez le AWS CLI et exécutez la commande suivante pour mettre à jour une cible afin d'inclure un nom et une description.

Linux & macOS

```
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 retourne des informations telles que les suivantes.

```
{
    "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). Veillez à spécifier un nœud supplémentaire configuré pour être utilisé avec Systems Manager.

Linux & macOS

```
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 retourne des informations telles que les suivantes.

```
{
    "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 & macOS

```
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 Run Command.

> **Tip**
>
> Si votre cible est une instance Amazon Elastic Compute Cloud (Amazon EC2) pour Windows Server, changez `df` à `ipconfig`, et `AWS-RunShellScript` à `AWS-RunPowerShellScript` dans la commande suivante.

Linux & macOS

```
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::account-id: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::account-id: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 retourne des informations telles que les suivantes.

```
{
    "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 & macOS

   ```
   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:region:111122223333:function:SSMTestLambda" \
       --service-role-arn "arn:aws:iam:account-id: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 --task-arn
    "arn:aws:lambda:region:111122223333:function:SSMTestLambda" ^
       --service-role-arn "arn:aws:iam:account-id: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 retourne des informations telles que les suivantes.

```
{
    "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 Step Functions, adaptez et exécutez la commande suivante pour la mettre à jour task-invocation-parameters.

   Linux & macOS

```
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:region:execution:SSMStepFunctionTest" \
    --service-role-arn "arn:aws:iam:account-id: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:region:execution:SSMStepFunctionTest" ^
    --service-role-arn "arn:aws:iam:account-id: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 retourne des informations telles que les suivantes.

```
{
    "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 toute tâche et si il est donc sécurisé d'annuler son enregistrement.

   Linux & macOS

   ```
   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 retourne des informations telles que les suivantes.

   ```
   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 & macOS

   ```
   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 retourne des informations telles que les suivantes.

```
{
    "WindowId": "mw-0c50858d01EXAMPLE",
    "WindowTargetId": "e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
}
```

9. Exécutez la commande suivante pour mettre à jour une tâche Run Command. Cet exemple utilise un paramètre Systems Manager Parameter Store appelé `UpdateLevel`, qui est formaté comme suit : `'{{ssm:UpdateLevel}}'`

   Linux & macOS

   ```
   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 retourne des informations telles que les suivantes.

```
{
    "WindowId": "mw-0c50858d01EXAMPLE",
    "WindowTaskId": "4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE",
    "Targets": [
        {
            "Key": "InstanceIds",
            "Values": [
                "i-02573cafcfEXAMPLE"
            ]
        }
    ],
    "TaskArn": "AWS-RunShellScript",
    "ServiceRoleArn": "arn:aws:iam::111122223333:role/
MyMaintenanceWindowServiceRole",
    "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 & macOS

    ```
    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:account-id:role/
MyMaintenanceWindowServiceRole \
    --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:account-id:role/
MyMaintenanceWindowServiceRole ^
    --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 retourne des informations telles que les suivantes.

```
{
    "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/
MyMaintenanceWindowServiceRole",
    "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 retourne des informations telles que les suivantes.

```
{
   "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 (AWS CLI) ou de la console Systems Manager. La fenêtre de maintenance que vous créez procède à la mise à jour de SSM Agent sur les nœuds gérés.

Table des matières

- Démonstration : création d'une fenêtre de maintenance pour l'application des correctifs (console)

Vous pouvez également consulter des exemples de commandes dans la Référence relative à la AWS CLI Systems Manager.

Procédure : Créer une fenêtre de maintenance pour mettre à jour l'SSM Agent (AWS CLI)

La procédure suivante vous montre comment utiliser l'AWS Command Line Interface (AWS CLI) pour créer une fenêtre de maintenance AWS Systems Manager. La démonstration explique également comment enregistrer des nœuds gérés en tant que cibles et comment enregistrer une tâche Systems Manager Run Command pour procéder à la mise à jour de SSM Agent.

Avant de commencer

Pour suivre la procédure ci-dessous, vous devez disposer d'autorisations administrateur sur les nœuds que vous souhaitez configurer, ou bénéficier des autorisations appropriées dans AWS Identity and Access Management (IAM). Vérifiez en outre que vous avez au moins un nœud exécuté géré pour Linux ou Windows Server configuré pour Systems Manager dans un environnement hybride et multicloud. Pour de plus amples informations, veuillez consulter Con AWS Systems Manager figuration.

Rubriques

- Étape 1 : Mise en route
- Étape 2 : Création de la fenêtre de maintenance
- Étape 3 : Enregistrement des cibles de fenêtre de maintenance (AWS CLI)
- Étape 4 : Enregistrement d'une tâche Run Command pour que la fenêtre de maintenance mette à jour SSM Agent

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 Command Line Interface (AWS CLI).

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l'AWS CLI.
2. Vérifiez qu'un nœud est prêt à être enregistré comme cible d'une fenêtre de maintenance.

Exécutez la commande suivante pour afficher les nœuds qui sont en ligne.

```
aws ssm describe-instance-information --query "InstanceInformationList[*]"
```

Exécutez la commande suivante pour afficher les détails relatifs à un nœud particulier.

```
aws ssm describe-instance-information --instance-information-filter-list
 key=InstanceIds,valueSet=instance-id
```

Étape 2 : Création de 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 2 h 00, dans le fuseau horaire États-Unis/Pacifique, avec une coupure d'une heure.

   Linux & macOS

   ```
   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`, consultez Référence : Expressions Cron et Rate pour Systems Manager.

Pour obtenir une explication des relations entre les différentes options liées à la planification pour les fenêtres de maintenance, consultez Options de planification de la fenêtre de maintenance et de période active.

Pour de plus amples informations sur l'utilisation de l'option `--schedule`, consultez Référence : Expressions Cron et Rate pour Systems Manager.

Le système retourne des informations telles que les suivantes.

```
{
   "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 retourne des informations telles que les suivantes.

```
{
    "WindowIdentities": [
        {
            "Cutoff": 1,
            "Name": "My-First-Maintenance-Window",
            "NextExecutionTime": "2019-02-03T02:00-08:00",
            "Enabled": true,
            "WindowId": "mw-0c50858d01EXAMPLE",
            "Duration": 2
        }
    ]
}
```

Étape 3 : Enregistrement des 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 nœuds à mettre à jour.

Pour enregistrer des cibles de fenêtre de maintenance (AWS CLI)

1. Exécutez la commande suivante. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   Linux & macOS

   ```
   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 valeur `WindowTargetId`. 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 nœuds gérés.

   Linux & macOS

   ```
   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 nœuds à l'aide de balises.

Linux & macOS

```
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 retourne des informations telles que les suivantes.

```
{
    "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 : Enregistrement d'une tâche Run Command pour que la fenêtre de maintenance mette à jour SSM Agent

Utilisez la procédure suivante pour enregistrer une tâche Run Command pour la fenêtre de maintenance que vous avez créée à l'étape 2. La tâche Run Command met à jour l'SSM Agent sur les cibles enregistrées.

Pour enregistrer une tâche Run Command pour une fenêtre de maintenance à mettre à jour SSM Agent (AWS CLI)

1. Exécutez la commande suivante pour enregistrer une tâche Run Command pour la fenêtre de maintenance à l'aide de la valeur `WindowTargetId` notée à l'étape 3. Remplacez chaque

example resource placeholder (espace réservé pour les ressources) avec vos propres informations. La tâche met à jour l'SSM Agent en utilisant le document `AWS-UpdateSSMAgent`.

Linux & macOS

```
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:account-id: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:account-id: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 retourne des informations telles que les suivantes.

```
{
   "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 retourne des informations telles que les suivantes.

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

Démonstration : Créer une fenêtre de maintenance pour mettre automatiquement à jour l'SSM Agent (console)

La procédure pas à pas suivante explique comment utiliser la AWS Systems Manager console pour créer une fenêtre de maintenance. La démonstration explique également comment enregistrer des nœuds gérés en tant que cibles et comment enregistrer une tâche Systems Manager Run Command pour procéder à la mise à jour de SSM Agent.

Avant de commencer

Avant de terminer la procédure suivante, vous devez disposer des autorisations d'administrateur sur les nœuds que vous souhaitez configurer ou avoir obtenu les autorisations appropriées dans AWS Identity and Access Management (IAM). Vérifiez en outre que vous avez au moins un nœud exécuté géré pour Linux ou Windows Server, dans un environnement hybride et multicloud, qui est configuré pour Systems Manager. Pour plus d'informations, consultez Con AWS Systems Manager figuration.

Rubriques

- Étape 1 : Création de la fenêtre de maintenance (console)
- Étape 2 : Enregistrement des cibles de fenêtre de maintenance (console)
- Étape 3 : Enregistrement d'une tâche Run Command pour que la fenêtre de maintenance mette à jour SSM Agent (console)

Étape 1 : Création de la fenêtre de maintenance (console)

Créer une fenêtre de maintenance (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Maintenance Windows.
3. Sélectionnez Create maintenance window (Créer une fenêtre de maintenance).
4. Pour Name (Nom), saisissez un nom évocateur pour vous aider à identifier cette fenêtre de maintenance.
5. (Facultatif) Sous Description, entrez une description.
6. Sélectionnez Allow unregistered targets (Autoriser les cibles non enregistrées) si vous souhaitez autoriser l'exécution d'une tâche de fenêtre de maintenance sur des nœuds gérés, même si vous n'avez pas enregistré ces nœuds comme cibles. Lorsque vous sélectionnez cette option, vous pouvez sélectionner les nœuds non enregistrés (par ID de nœud) lorsque vous enregistrez une tâche auprès de la fenêtre de maintenance.

   Si vous ne sélectionnez 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.

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 Window start date (optional) [Fenêtre de 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.

> **Note**
>
> Vous ne pouvez pas spécifier une date et une heure de début antérieures.

11. (Facultatif) Pour Window start date (optional) [Fenêtre de 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 Schedule 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). Par 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. Sélectionnez 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 Enabled (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 nœuds 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. Sélectionnez Actions, puis Register targets (Enregistrer les cibles).
3. (Facultatif) Pour Target Name (Nom de cible), saisissez un nom pour la cible.
4. (Facultatif) Sous 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 EventBridge événement Amazon déclenché lors de l'exécution de tâches pour ces cibles dans cette fenêtre de maintenance.

   Pour plus d'informations sur l'utilisation EventBridge pour surveiller les événements de Systems Manager, consultezSurveillance d'événements Systems Manager avec Amazon EventBridge.
6. Dans la zone Targets (Cibles), sélectionnez l'une des options décrites dans le tableau suivant.

Option	Description
Spécification de balises d'instance	Dans les zones Specify instance tags (Spécifier les balises d'instance), spécifiez une ou plusieurs clés de balise et (facultat if) des valeurs de balise qui ont été ou seront ajoutées aux nœuds gérés de votre

Option	Description
	compte. Lorsqu'elle s'exécute, la fenêtre de maintenance tente d'exécuter des tâches sur tous les nœuds gérés auxquels ces balises ont été ajoutées. Si vous spécifiez plusieurs clés de balise, un nœud doit être balisé avec toutes les clés et valeurs de balise que vous décidez d'inclure dans le groupe cible.
Choix manuel des nœuds	Dans la liste, cochez la case située en regard de chaque nœud que vous souhaitez inclure dans la cible de fenêtre de maintenance. La liste inclut tous les nœuds de votre compte qui sont configurés pour être utilisés avec Systems Manager. Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés pour obtenir des conseils de dépannage. Pour les appareils de périphérie, les serveurs sur site et les machines virtuelles (VM), consultez Utilisation de Systems Manager dans des environnements hybrides et multicloud.

Option	Description
Pour choisir un groupe de ressources	Pour Resource group (Groupe de ressources), sélectionnez 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 : • Que sont les groupes de ressources ? dans le Guide de l'utilisateur AWS Resource Groups • Groupes de ressources et balisage pour AWS dans l'AWS News Blog. Pour Resource types (Types de ressource), sélectionnez jusqu'à cinq types de ressources disponibles, ou sélectionnez 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`

Option	Description
	Mais plus tard, lorsque vous ajoutez des tâches à la fenêtre de maintenance, vous incluez uniquement des tâches qui exécutent des actions sur les nœuds, comme l'application d'un référentiel de correctifs ou le redémarrage d'un nœud. Dans le journal de fenêtre de maintenance, une erreur peut signaler qu'aucun compartiment Amazon Simple Storage Service (Amazon S3) ou table Amazon DynamoDB n'a été trouvé. Toutefois, la fenêtre de maintenance continue d'exécuter des tâches sur les nœuds de votre groupe de ressources.

7. Sélectionnez Register target (Enregistrer la cible).

Étape 3 : Enregistrement d'une tâche Run Command pour que la fenêtre de maintenance mette à jour SSM Agent (console)

Utilisez la procédure suivante pour enregistrer une tâche Run Command pour la fenêtre de maintenance que vous avez créée à l'étape 1. La tâche Run Command met à jour l'SSM Agent 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. Sélectionnez Actions, puis Register Run Command task (Enregistrer une tâche d'exécution de commande).
3. (Facultatif) Pour Name (Nom), saisissez un nom pour la tâche, tel qu'UpdateSSMAgent.
4. (Facultatif) Sous Description, entrez une description.
5. Dans la zone Document de commande, sélectionnez 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 Version du document, sélectionnez la version de document à utiliser.
7. 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.
8. Dans la section Targets (Cibles), identifiez les nœuds sur lesquels vous souhaitez exécuter cette opération en choisissant Selecting registered target groups (Sélectionner des groupes cibles enregistrés) ou Selecting unregistered targets (Sélectionner des cibles non enregistrées).
9. Pour Rate control (Contrôle de débit) :

   - Dans Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage de nœuds gérés sur lesquels exécuter simultanément la commande.

     > **Note**
     >
     > Si vous avez sélectionné des cibles en spécifiant les balises appliquées aux nœuds gérés ou en spécifiant des groupes de ressources AWS , et que vous n'êtes pas certain du nombre de nœuds gérés ciblés, limitez le nombre de cibles autorisées à exécuter simultanément le document en indiquant un pourcentage.

   - Dans Error threshold (Seuil d'erreur), indiquez quand arrêter l'exécution de la commande sur les autres nœuds gérés après l'échec de celle-ci sur un certain nombre ou un certain pourcentage de nœuds. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les nœuds gérés sur lesquels la commande est toujours en cours de traitement peuvent également envoyer des erreurs.
10. (Facultatif) Pour le rôle de service IAM, choisissez un rôle qui fournira des autorisations à Systems Manager lors de l'exécution d'une tâche pendant la fenêtre de maintenance.

    Si vous ne spécifiez aucun ARN de rôle de service, Systems Manager utilise un rôle lié à un service dans votre compte. S'il n'existe aucun rôle lié à un service approprié pour Systems Manager dans votre compte, il est créé lorsque la tâche est enregistrée avec succès.

> **Note**
>
> Pour améliorer le niveau de sécurité, nous vous recommandons vivement de créer une politique personnalisée et un rôle de service personnalisé pour exécuter les tâches de votre fenêtre de maintenance. La politique peut être conçue pour fournir uniquement les autorisations nécessaires pour les tâches spécifiques de votre fenêtre de maintenance. Pour plus d'informations, consultez Utiliser la console pour configurer les autorisations pour les fenêtres de maintenance.

11. (Facultatif) Pour Options de sortie, procédez de l'une des manières suivantes :

    - Cochez la case Activer l'écriture dans S3 pour enregistrer la sortie de la commande dans un fichier. Saisissez les noms de compartiment et de préfixe (dossier) dans les zones.

      > **Note**
      >
      > Les autorisations S3 qui donnent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué au nœud, et non celles de l'utilisateur qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager. En outre, si le compartiment S3 spécifié se trouve sur un autre Compte AWS, vérifiez que le profil d'instance associé au nœud dispose des autorisations nécessaires pour écrire dans ce compartiment.

    - Cochez la case CloudWatch de sortie pour écrire la sortie complète dans Amazon CloudWatch Logs. Entrez le nom d'un groupe de CloudWatch journaux Logs.

12. Dans la section Notifications SNS, vous pouvez éventuellement autoriser Systems Manager à envoyer des notifications à propos des statuts de commande en utilisant Amazon Simple Notification Service (Amazon SNS). Si vous choisissez d'activer cette option, vous devez spécifier les informations suivantes :

    a. Le rôle IAM pour démarrer 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 nœuds, sélectionnez Invocation

(Appel) pour recevoir une notification par appel (par nœud) lors du changement de statut de chaque appel.

13. Dans la section Parameters (Paramètres), vous pouvez éventuellement fournir une version spécifique de l'SSM Agent à installer, ou vous pouvez autoriser la rétrogradation du service SSM Agent à une version antérieure. Toutefois, pour cette procédure pas à pas, nous n'indiquons aucune version. Par conséquent, l'SSM Agent est mis à jour vers la dernière version.
14. Sélectionnez Register run command task (Enregistrer une tâche d'exécution de commande).

Démonstration : création d'une fenêtre de maintenance pour l'application des correctifs (console)

> **Important**
>
> Vous pouvez continuer à utiliser cette rubrique existante pour créer une fenêtre de maintenance afin d'appliquer des correctifs. Toutefois, nous vous recommandons plutôt d'utiliser une politique de correctifs. Pour plus d'informations, consultez Utilisation des stratégies de correctifs Quick Setup et Configuration des correctifs de l'organisation Patch Manager.

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, consultez AWS Systems Manager Maintenance Windows.

Vous devez configurer les rôles et les autorisations pourMaintenance Windows, une fonctionnalité de AWS Systems Manager, avant de commencer cette procédure. Pour plus d'informations, consultez Configuration de Maintenance Windows.

Pour créer une fenêtre de maintenance pour l'application des correctifs

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Maintenance Windows.
3. Sélectionnez 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. Sélectionnez Allow unregistered targets (Autoriser les cibles non enregistrées) si vous souhaitez autoriser l'exécution d'une tâche de fenêtre de maintenance sur des nœuds gérés, même si vous n'avez pas enregistré ces nœuds comme cibles. Lorsque vous sélectionnez cette option, vous pouvez sélectionner les nœuds non enregistrés (par ID de nœud) lorsque vous enregistrez une tâche auprès de la fenêtre de maintenance.

   Si vous ne sélectionnez 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.
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). Par 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. Sélectionnez Create maintenance window (Créer une fenêtre de maintenance).
14. Dans la liste des fenêtres de maintenance, sélectionnez la fenêtre de maintenance que vous venez de créer, puis sélectionnez 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), sélectionnez Specifying instance tags (Spécification des balises d'instance).
17. Dans Instance tags (Balises d'instance), saisissez une clé et une valeur de balise pour identifier les nœuds à enregistrer auprès de la fenêtre de maintenance, puis sélectionnez Add (Ajouter).
18. Sélectionnez 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, sélectionnez 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), sélectionnez `AWS-RunPatchBaseline`.
22. Pour Task priority (Priorité de tâche), sélectionnez une priorité. Zéro (`0`) est la priorité la plus élevée.
23. Dans Targets (Cibles), sous Target by (Cible par), sélectionnez la cible de fenêtre de maintenance que vous avez créée précédemment dans cette procédure.
24. Pour Rate control (Contrôle de débit) :

    - Dans Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage de nœuds gérés sur lesquels exécuter simultanément la commande.

      > **Note**
      >
      > Si vous avez sélectionné des cibles en spécifiant les balises appliquées aux nœuds gérés ou en spécifiant des groupes de ressources AWS , et que vous n'êtes pas

certain du nombre de nœuds gérés ciblés, limitez le nombre de cibles autorisées à exécuter simultanément le document en indiquant un pourcentage.

- Dans Error threshold (Seuil d'erreur), indiquez quand arrêter l'exécution de la commande sur les autres nœuds gérés après l'échec de celle-ci sur un certain nombre ou un certain pourcentage de nœuds. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les nœuds gérés sur lesquels la commande est toujours en cours de traitement peuvent également envoyer des erreurs.

25. (Facultatif) Pour le rôle de service IAM, choisissez un rôle qui fournira des autorisations à Systems Manager lors de l'exécution d'une tâche pendant la fenêtre de maintenance.

    Si vous ne spécifiez aucun ARN de rôle de service, Systems Manager utilise un rôle lié à un service dans votre compte. S'il n'existe aucun rôle lié à un service approprié pour Systems Manager dans votre compte, il est créé lorsque la tâche est enregistrée avec succès.

    > **Note**
    >
    > Pour améliorer le niveau de sécurité, nous vous recommandons vivement de créer une politique personnalisée et un rôle de service personnalisé pour exécuter les tâches de votre fenêtre de maintenance. La politique peut être conçue pour fournir uniquement les autorisations nécessaires pour les tâches spécifiques de votre fenêtre de maintenance. Pour plus d'informations, consultez Utiliser la console pour configurer les autorisations pour les fenêtres de maintenance.

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) Saisissez les noms de compartiment et de préfixe (dossier) dans les zones.

    > **Note**
    >
    > Les autorisations S3 qui donnent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué au nœud géré, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, vérifiez que le profil d'instance ou le rôle de

service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

Pour diffuser la sortie vers un groupe de CloudWatch journaux Amazon Logs, cochez la case CloudWatch de sortie. Saisissez le nom du groupe de journaux dans la zone.

27. Dans la section SNS notifications (Notifications SNS), si vous souhaitez envoyer des notifications sur le statut d'exécution des commandes, cochez la case Enable SNS notifications (Activer les notifications SNS).

    Pour plus d'informations sur la configuration des notifications Amazon SNS pour Run Command, consultez Surveillance des changements d'état du Systems Manager à l'aide des notifications Amazon SNS.

28. Pour Parameters (Paramètres) :

    - Pour Operation (Opération), sélectionnez Scan (Analyser) afin de rechercher les correctifs manquants, ou sélectionnez 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 de saisir quoi que ce soit dans le champ Install Override List (Liste de remplacement d'installation) sauf si vous souhaitez que Patch Manager utilise un jeu de correctifs différent de celui spécifié pour le référentiel de correctifs. Pour plus d'informations, consultez Nom du paramètre: `InstallOverrideList`.
    - Dans Reboot option (Option de redémarrage), indiquez si vous souhaitez que les nœuds redémarrent si des correctifs sont installés pendant l'opération `Install`, ou si Patch Manager détecte que d'autres correctifs ont été installés depuis le dernier redémarrage du nœud. Pour plus d'informations, consultez Nom du paramètre: `RebootOption`.
    - (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. Sélectionnez 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 explorer un nœud spécifique en le sélectionnant sur la page Managed Instances (Instances gérées), puis en choisissant l'onglet Patch (Correctif). Vous pouvez également utiliser les API DescribePatchGroupStateet DescribeInstancePatchStatesForPatchGroup pour consulter les détails de conformité. Pour plus d'informations sur les données de conformité des correctifs, consultez A propos de la conformité des correctifs.

À propos des planifications d'application des correctifs en utilisant des fenêtres de maintenance

Une fois que vous avez configuré un référentiel de correctifs (et éventuellement un groupe de correctifs), vous pouvez appliquer des correctifs à votre nœud 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 indiquant la `Patch Group` ou `PatchGroup` balise pour le nom de celle-ci, et toute valeur pour laquelle vous avez défini des balises Amazon Elastic Compute Cloud (Amazon EC2), par exemple, "serveurs web" ou "US-EAST-PROD". (Utilisez `PatchGroup`, sans espace, si vous avez autorisé les balises dans les métadonnées d'instance EC2..
3. Créez une nouvelle tâche de fenêtre de maintenance et spécifiez le document `AWS-RunPatchBaseline`.

Lorsque vous configurez la tâche, vous pouvez choisir de procéder à une simple analyse des nœuds, ou bien de les analyser et d'y installer les correctifs. Si vous choisissez de procéder à une simple analyse des nœuds, la fonctionnalité Patch Manager d'AWS Systems Manager analyse chaque nœud et génère une liste de correctifs manquants que vous devez passer en revue.

Si vous choisissez d'analyser les nœuds et d'y installer les correctifs, Patch Manager analyse chaque nœud et compare la liste des correctifs installés avec celle des correctifs approuvés dans le référentiel. Patch Manager 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 Run Command afin d'appeler directement le document `AWS-RunPatchBaseline`.

> **Important**
>
> Une fois les correctifs installés, Systems Manager redémarre chaque nœud. Ce redémarrage est nécessaire pour vérifier que les correctifs sont correctement installés et que le système n'a pas laissé le nœud dans un état potentiellement incorrect. (Exception : si le paramètre `RebootOption` est défini sur `NoReboot` dans le document `AWS-RunPatchBaseline`, le nœud géré n'est pas redémarré après l'exécution de Patch Manager. Pour plus d'informations, consultez Nom du paramètre: `RebootOption`).

Utilisation de pseudo-paramètres lors de l'enregistrement des tâches de la fenêtre de maintenance

Lorsque vous enregistrez une tâche dans Maintenance Windows une fonctionnalité de AWS Systems Manager, vous spécifiez les paramètres uniques à chacun des quatre types de tâches. (Dans les commandes CLI, celles-ci sont fournies à l'aide de l'`--task-invocation-parameters`option.)

Vous pouvez également référencer certaines valeurs en utilisant la syntaxe du pseudo-paramètre, comme `{{RESOURCE_ID}}`, `{{TARGET_TYPE}}` et `{{WINDOW_TARGET_ID}}`. Une fois que la tâche de la 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 plus loin dans cette rubrique dansPseudo-paramètres pris en charge.

> **Important**
>
> Pour le type de cible `RESOURCE_GROUP`, selon le format d'ID requis pour la tâche, vous pouvez choisir entre utiliser `{{TARGET_ID}}` et `{{RESOURCE_ID}}` pour référencer la ressource lors de l'exécution de la tâche. `{{TARGET_ID}}` renvoie l'ARN complet de la

ressource. {{RESOURCE_ID}} renvoie uniquement un nom ou un ID plus court de la ressource, comme indiqué dans ces exemples.

- Format {{TARGET_ID}} : arn:aws:ec2:us-east-1:123456789012:instance/i-02573cafcfEXAMPLE
- Format {{RESOURCE_ID}} : 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 Pseudo-paramètres pris en charge.
{{TARGET_ID}}et {{RESOURCE_ID}} peut être utilisé pour transmettre des identifiants de AWS ressources uniquement aux tâches Automation, Lambda et Step Functions. Ces deux pseudo-paramètres ne peuvent pas être utilisés avec des tâches Run Command.

Exemples de pseudo-paramètres

Supposons que votre charge utile pour une AWS Lambda tâche 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}}. Par exemple :

```
"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 Amazon Elastic Compute Cloud (Amazon EC2), par exemple une table Amazon 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 afficher la liste des types de ressources pris en charge, consultez Pseudo-paramètres pris en charge.

Autre exemple, pour exécuter une tâche Automation qui arrête vos instances EC2, vous spécifiez le document Systems Manager (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 Automation qui copie un instantané d'un volume Amazon Elastic Block Store (Amazon EBS), spécifiez le document SSM `AWS-CopySnapshot` 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 la fenêtre qui inclut la cible (ID de cible).
- **`WINDOW_EXECUTION_ID`** : ID de l'exécution de fenêtre actuelle.
- **`TASK_EXECUTION_ID`** : ID de l'exécution de tâche actuelle.
- **`INVOCATION_ID`** : ID de l'appel actuel.
- **`TARGET_TYPE`** : type de la 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 valeurs TARGET_ID pour des types de ressources particuliers d'un groupe de ressources.

> **Note**
>
> TARGET_ID n'est pas pris en charge pour des tâches Run Command.

Type de ressource	Exemple TARGET_ID
AWS::CloudWatch::Alarm	arn:aws:cloudwatch:us-east-1:123456789012:alarm:MyCloudWatchAlarm i-078a280217EXAMPLE
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
AWS::EC2::Snapshot	arn:aws:ec2:us-east-1:123456789012:snapshot/snap-03866bf003EXAMPLE

Type de ressource	Exemple TARGET_ID
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::DBCluster	arn:aws:rds:us-east-2:123456789012:cluster:My-Cluster
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.

> **Note**
>
> RESOURCE_ID n'est pas pris en charge pour des tâches Run Command.

Type de ressource	Exemple RESOURCE_ID	
AWS::CloudWatch::Alarm	MyCloudWatchAlarm	
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::DBCluster	My-Cluster	
AWS::RDS::DBInstance	My-SQL-Instance	
AWS::S3::Bucket	DOC-EXAMPLE-BUCKET	
AWS::SSM::ManagedInstance	mi-0feadcf2d9EXAMPLE	

Note

Si le groupe de AWS ressources que vous spécifiez inclut des types de ressources qui ne génèrent aucune `RESOURCE_ID` valeur et ne sont pas répertoriés dans le tableau précédent, le `RESOURCE_ID` paramètre 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.

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. 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 et 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). Par exemple : `America/Chicago`, `Europe/Berlin` ou `Asia/Tokyo`

Exemples

- Exemple 1 : Spécification d'une date de début de fenêtre de maintenance
- Exemple 2 : Spécification de la date de début et de la date de fin d'une fenêtre de maintenance
- Exemple 3 : Création d'une fenêtre de maintenance ne s'exécutant qu'une seule fois
- Exemple 4 : Spécifiez le nombre de jours de décalage programmé pour une fenêtre de maintenance

Exemple 1 : Spécification d'une date de début de fenêtre de maintenance

Imaginons que vous utilisiez l'AWS Command Line Interface (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 *)"`

Par exemple :

Linux & macOS

```
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 n'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.) 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 valeurs ensemble, les valeurs `--schedule-timezone` et `--schedule` signifient que la fenêtre de maintenance démarrera à 9 heures tous les mercredis dans le fuseau horaire États-Unis/Pacifique (représenté par « America/Los Angeles » en format IANA). La première exécution durant la période d'activité aura lieu le mercredi 4 janvier 2021, à 9 heures, heure des États-Unis/Pacifique.

Exemple 2 : Spécification de la date de début et de la date de fin d'une 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)"`

Par exemple :

Linux & macOS

```
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 à neuf heures d'avance par rapport à l'heure UTC.) Si l'on considère les deux valeurs ensemble, les valeurs `--schedule-timezone` et `--schedule` signifient que la fenêtre de maintenance s'exécutera à 3 h 15 tous les mardis dans le fuseau horaire standard du Japon (représenté par « Asia/Tokyo » en 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 : Création d'une fenêtre de maintenance ne s'exécutant qu'une seule fois

Vous avez à présent créé une fenêtre de maintenance avec cette option :

- `--schedule "at(2020-07-07T15:55:00)"`

Par exemple :

Linux & macOS

```
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 : Spécifiez 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
```

Par exemple :

Linux & macOS

```
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.

Plus d'informations

- Référence : Expressions Cron et Rate pour Systems Manager
- Créer une fenêtre de maintenance (console)
- Didacticiel : Créer et configurer une fenêtre de maintenance (AWS CLI)
- CreateMaintenanceWindow dans la Référence d'API AWS Systems Manager
- create-maintenance-window dans la section AWS Systems Manager de la Référence des commandes AWS CLI
- Time Zone Database sur le site web de l'IANA

Enregistrement de tâches de fenêtre de maintenance sans cibles

Pour chaque fenêtre de maintenance créée, vous pouvez spécifier une ou plusieurs tâches à effectuer lors de l'exécution de la fenêtre de maintenance. Dans la plupart des cas, vous devez spécifier les ressources, ou cibles, sur lesquelles la tâche doit s'exécuter. Dans certains cas, cependant, vous n'avez pas à spécifier explicitement des cibles dans la tâche.

Une ou plusieurs cibles doivent être spécifiées pour des tâches de fenêtre de maintenance de type Systems Manager Run Command. En fonction de la nature de la tâche, les cibles sont facultatives pour d'autres types de tâches de fenêtre de maintenance (Automation Systems Manager, AWS Lambda et AWS Step Functions).

Pour les types de tâches Lambda et Step Functions, la nécessité d'une cible dépend du contenu de la fonction ou de la machine d'état que vous avez créée.

Dans bien des cas, vous n'avez pas à spécifier explicitement une cible pour une tâche d'automatisation. Par exemple, supposons que vous créiez une tâche de type Automation pour mettre à jour une Amazon Machine Image (AMI) pour Linux à l'aide du runbook `AWS-UpdateLinuxAmi`. Lorsque la tâche s'exécute, l'AMI est mise à jour avec les derniers packages de distribution Linux et les logiciels Amazon disponibles. Ces mises à jour sont déjà installées sur les nouvelles instances créées à partir de l'AMI. Comme l'ID de l'AMI à mettre à jour est spécifié dans les paramètres d'entrée du runbook, il est inutile de spécifier à nouveau une cible dans la tâche de la fenêtre de maintenance.

De même, supposons que vous utilisiez l'AWS Command Line Interface (AWS CLI) pour enregistrer une tâche Automation de fenêtre de maintenance utilisant le runbook `AWS-RestartEC2Instance`.

Comme le nœud à redémarrer est spécifié dans l'argument `--task-invocation-parameters`, l'option `--targets` n'est pas nécessaire.

> **Note**
>
> Pour les tâches de la fenêtre de maintenance qui n'ont pas de cible spécifiée, vous ne pouvez pas fournir de valeurs pour `--max-errors` et `--max-concurrency`. Au lieu de cela, le système insère une valeur d'espace réservé de 1, qui peut être rapportée dans la réponse à des commandes telles que describe-maintenance-window-tasks et get-maintenance-window-task. Ces valeurs n'affectent pas l'exécution de votre tâche et peuvent être ignorées.

Les exemples suivants montrent également que les options `--targets`, `--max-errors` et `--max-concurrency` peuvent être omises pour une tâche de fenêtre de maintenance sans cible.

Linux & macOS

```
aws ssm register-task-with-maintenance-window \
    --window-id "mw-ab12cd34eEXAMPLE" \
    --service-role-arn "arn:aws:iam::123456789012:role/
MaintenanceWindowAndAutomationRole" \
    --task-type "AUTOMATION" \
    --name "RestartInstanceWithoutTarget" \
    --task-arn "AWS-RestartEC2Instance" \
    --task-invocation-parameters "{\"Automation\":{\"Parameters\":{\"InstanceId\":
[\"i-02573cafcfEXAMPLE\"]}}}" \
    --priority 10
```

Windows

```
aws ssm register-task-with-maintenance-window ^
    --window-id "mw-ab12cd34eEXAMPLE" ^
    --service-role-arn "arn:aws:iam::123456789012:role/
MaintenanceWindowAndAutomationRole" ^
    --task-type "AUTOMATION" ^
    --name "RestartInstanceWithoutTarget" ^
    --task-arn "AWS-RestartEC2Instance" ^
    --task-invocation-parameters "{\"Automation\":{\"Parameters\":{\"InstanceId\":
[\"i-02573cafcfEXAMPLE\"]}}}" ^
    --priority 10
```

> **Note**
>
> Pour les tâches de fenêtre de maintenance enregistrées avant le 23 décembre 2020 : si vous avez spécifié des cibles pour la tâche et que l'une d'entre elles n'est plus requise, vous pouvez mettre à jour cette tâche afin de supprimer les cibles en utilisant la console Systems Manager ou la commande AWS CLI update-maintenance-window-task.

Plus d'informations

- Messages d'erreur : « Les tâches de la fenêtre de maintenance sans cibles ne prennent pas en charge MaxConcurrency les valeurs » et « Les tâches de la fenêtre de maintenance sans cibles ne prennent pas en charge MaxErrors les valeurs »

Résolution des problèmes liés aux fenêtres de maintenance

Consultez les informations suivantes pour tenter de résoudre les problèmes liés aux fenêtres de maintenance.

Rubriques

- Erreur de modification de tâche : sur la page de modification de tâche de fenêtre de maintenance, la liste de rôles IAM affiche 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.)
- Toutes les cibles de fenêtre de maintenance ne sont pas mises à jour
- La tâche échoue avec le statut d'invocation de tâche : « The provided role does not contain the correct SSM permissions. » (Le rôle fourni ne contient pas les autorisations SSM correctes).
- La tâche échoue avec le message d'erreur : « L'étape échoue lorsqu'elle valide et résout les entrées de l'étape »
- Messages d'erreur : « Les tâches de la fenêtre de maintenance sans cibles ne prennent pas en charge MaxConcurrency les valeurs » et « Les tâches de la fenêtre de maintenance sans cibles ne prennent pas en charge MaxErrors les valeurs »

Erreur de modification de tâche : sur la page de modification de tâche de fenêtre de maintenance, la liste de rôles IAM affiche 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 de fenêtre de maintenance AWS Identity and Access Management (IAM) que vous avez indiqué initialement a été supprimé après avoir créé la tâche.

Correctifs possibles : 1) sélectionnez un autre rôle de fenêtre de maintenance IAM, s'il existe pour votre compte, ou créez-en un nouveau et sélectionnez-le pour cette tâche.

Problème 2 : si la tâche a été créée à l'aide de l'AWS Command Line Interface (AWS CLI), de AWS Tools for Windows PowerShell ou d'un kit SDK AWS, un nom de rôle de fenêtre de maintenance IAM inexistant pourrait avoir été spécifié. Par exemple, il est possible que le rôle de fenêtre de maintenance 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 : sélectionnez le nom correct du rôle de fenêtre de maintenance IAM que vous souhaitez utiliser, ou créez-en un nouveau à spécifier pour cette tâche.

Toutes les cibles de fenêtre de maintenance ne sont pas mises à jour

Problème : vous remarquez que des tâches de fenêtre de maintenance ne se sont pas 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 de cette ressource est marquée comme ayant échoué ou expiré.

Solution: les raisons les plus courantes pour lesquelles une tâche de fenêtre de maintenance ne s'exécute pas sur une ressource cible concernent la connectivité et la disponibilité. Par exemple :

- Systems Manager a perdu la connexion avec la ressource avant ou pendant l'opération de la fenêtre de maintenance.
- La ressource était hors ligne ou arrêtée pendant l'opération de la 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 fenêtre de maintenance sur les ressources qui n'étaient pas disponibles ou qui étaient hors connexion.

La tâche échoue avec le statut d'invocation de tâche : « The provided role does not contain the correct SSM permissions. » (Le rôle fourni ne contient pas les autorisations SSM correctes).

Problème : Vous avez spécifié une fonction du service de fenêtre de maintenance pour une tâche, mais la tâche ne s'exécute pas correctement et le statut d'invocation de la tâche indique que « The provided role does not contain the correct SSM permissions. » (Le rôle fourni ne contient pas les autorisations SSM correctes).

- Solution : Dans Tâche 1 : création d'une politique pour votre fonction du service de fenêtre d'entretien personnalisée, nous fournissons une politique de base que vous pouvez attacher à votre fonction du service de fenêtre de maintenance personnalisée. Cette politique comprend les autorisations nécessaires pour de nombreux scénarios de tâches. Cependant, en raison de la grande variété de tâches que vous pouvez exécuter, vous devrez peut-être fournir des autorisations supplémentaires dans la politique pour votre rôle de fenêtre de maintenance.

  Par exemple, certaines actions Automation utilisent des piles AWS CloudFormation. Par conséquent, vous devrez peut-être ajouter les autorisations supplémentaires `cloudformation:CreateStack`, `cloudformation:DescribeStacks`, et `cloudformation:DeleteStack` à la politique pour votre fonction du service de fenêtre de maintenance.

  Autre exemple, le runbook d'automatisation `AWS-CopySnapshot` requiert des autorisations pour créer un instantané Amazon Elastic Block Store (Amazon EBS). Par conséquent, vous devrez peut-être ajouter l'autorisation `ec2:CreateSnapshot`.

  Pour plus d'informations sur les autorisations de rôle requises par un runbook d'automatisation géré par AWS, consultez les descriptions de runbook dans la référence du runbook d'automatisation d'AWS Systems Manager.

  Pour plus d'informations sur les autorisations de rôle requises par un document SSM géré par AWS, consultez le contenu du document dans la section Documents de la console Systems Manager.

Pour plus d'informations sur les autorisations de rôle nécessaires pour les tâches Step Functions, les tâches Lambda, les runbooks d'automatisation personnalisés et les documents SSM, vérifiez les autorisations requises auprès de l'auteur de ces ressources.

La tâche échoue avec le message d'erreur : « L'étape échoue lorsqu'elle valide et résout les entrées de l'étape »

Problème : un runbook Automation ou un document de commande Systems Manager utilisé dans une tâche exige que vous spécifiez des entrées telles que `InstanceId` ou `SnapshotId`, mais une valeur n'est pas fournie ou n'est pas fournie correctement.

- Solution 1 : si votre tâche ne cible qu'une seule ressource, telle qu'un nœud ou un instantané, saisissez son ID dans les paramètres d'entrée de la tâche.
- Solution 2 : si votre tâche cible plusieurs ressources, comme la création d'images à partir de plusieurs nœuds lorsque vous utilisez le runbook `AWS-CreateImage`, vous pouvez utiliser l'un des pseudo-paramètres pris en charge pour les tâches de fenêtre de maintenance dans les paramètres d'entrée afin de représenter les ID des nœuds dans la commande.

  Les commandes suivantes enregistrent une tâche Systems Manager Automation auprès d'une fenêtre de maintenance en utilisant la AWS CLI. La valeur `--targets` indique un ID de cible de fenêtre de maintenance. En outre, même si le paramètre `--targets` spécifie un ID de cible de fenêtre, les paramètres du runbook Automation exigent qu'un ID de nœud soit fourni. Dans ce cas, la commande utilise le pseudo paramètre `{{RESOURCE_ID}}` en tant que la valeur `InstanceId`.

  Commande de l'AWS CLI :

  Linux & macOS

  L'exemple de commande suivant redémarre les instances Amazon Elastic Compute Cloud (Amazon EC2) appartenant au groupe cible de fenêtre de maintenance portant 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='{{RESOURCE_ID}}'}}" ^
    --priority 0 --max-concurrency 10 --max-errors 5 --name "My-Restart-EC2-
Instances-Automation-Task" ^
    --description "Automation task to restart EC2 instances"
```

Pour de plus amples informations sur l'utilisation de pseudo-paramètres pour les tâches de fenêtre de maintenance, veuillez consulter Utilisation de pseudo-paramètres lors de l'enregistrement des tâches de la fenêtre de maintenance et Exemples d'enregistrement de tâches.

Messages d'erreur : « Les tâches de la fenêtre de maintenance sans cibles ne prennent pas en charge MaxConcurrency les valeurs » et « Les tâches de la fenêtre de maintenance sans cibles ne prennent pas en charge MaxErrors les valeurs »

Problème : lorsque vous enregistrez une tâche de type Run Command, vous devez spécifier au moins une cible sur laquelle la tâche doit s'exécuter. Pour les autres types de tâches (Automation, AWS Lambda et AWS Step Functions), selon la nature de la tâche, les cibles sont facultatives. Les options `MaxConcurrency` (le nombre de ressources sur lesquelles exécuter une tâche simultanément) et `MaxErrors` (le nombre d'échecs d'exécution de la tâche sur les ressources cibles avant que la tâche n'échoue) ne sont ni requises, ni prises en charge pour les tâches de fenêtre de maintenance qui ne spécifient pas de cibles. Le système génère ces messages d'erreur si des valeurs sont spécifiées pour l'une ou l'autre de ces options alors qu'aucune cible de tâche n'est spécifiée.

Solution : si vous recevez l'une de ces erreurs, supprimez les valeurs de concomitance et de seuil d'erreur avant de poursuivre l'enregistrement ou de mettre à jour la tâche de fenêtre de maintenance.

Pour de plus amples informations sur l'exécution des tâches qui ne spécifient pas de cibles, veuillez consulter Enregistrement de tâches de fenêtre de maintenance sans cibles dans le Guide de l'utilisateur AWS Systems Manager.

AWS Systems Manager Gestion des nœuds

AWS Systems Manager fournit les fonctionnalités suivantes pour accéder, gérer et configurer vos nœuds gérés. Un nœud géré est une machine configurée pour être utilisée avec Systems Manager dans un environnement hybride et multicloud.

Rubriques

AWS Systems Manager Fleet Manager

Fleet Manager, une fonctionnalité de AWS Systems Manager, est une expérience d'interface utilisateur (UI) unifiée qui vous aide à gérer à distance vos nœuds exécutés sur site AWS ou sur site. Avec Fleet Manager, vous pouvez consulter l'état et le statut de performance de votre flotte de serveurs à partir d'une console unique. Vous pouvez également collecter des données provenant de nœuds individuels pour accomplir des tâches courantes de résolution des problèmes et de gestion à partir de la console. Cela inclut la connexion aux instances Windows à l'aide du protocole RDP (Remote Desktop Protocol), l'affichage du contenu des dossiers et des fichiers, la gestion du registre Windows, la gestion des utilisateurs du système d'exploitation, etc. Pour vos premiers pas dans Fleet Manager, ouvrez Systems Manager console. Dans le panneau de navigation, sélectionnez Fleet Manager.

À qui est destiné Fleet Manager ?

Tout AWS client qui souhaite une méthode centralisée pour gérer son parc de nœuds doit l'utiliserFleet Manager.

Comment mon organisation peut-elle tirer parti de Fleet Manager ?

Fleet Manager offre les avantages suivants :

- Accomplissez différentes tâches courantes d'administration des systèmes sans avoir à vous connecter manuellement à vos nœuds gérés.
- Gérez les nœuds exécutés sur plusieurs plateformes à partir d'une seule console unifiée.
- Gérez les nœuds exécutant différents systèmes d'exploitation à partir d'une seule console unifiée.
- Améliorez l'efficacité de l'administration de vos systèmes.

Quelles sont les fonctions d'Fleet Manager ?

Les principales fonctionnalités de Fleet Manager sont décrites ci-après.

- Accès au portail de la base de connaissances Red Hat

  Accédez aux binaires, ainsi qu'aux forums de partage de connaissances et de discussion disponibles sur le portail de la base de connaissances Red Hat via vos instances Red Hat Enterprise Linux (RHEL).
- État des nœuds gérés

  Identifiez les instances gérées qui sont `running` et celles qui sont `stopped`. Pour plus d'informations sur les instances arrêtées, consultez Arrêter et démarrer votre instance dans le guide de l'utilisateur Amazon EC2. Pour les appareils AWS IoT Greengrass principaux, vous pouvez voir lesquels sont `online` ou afficher un état de`Connection lost`. `offline`

  > **Note**
  >
  > Si vous avez arrêté votre instance gérée avant le 12 juillet 2021, elle n'affichera pas le marqueur `stopped`. Pour afficher le marqueur, démarrez et arrêtez l'instance.
- Affichage des informations d'instance

  Affichez des informations sur les données de dossier et de fichier stockées sur les volumes attachés à vos instances gérées, les données de performances relatives à vos instances en temps réel et les données de journal stockées sur vos instances.
- Affichage des informations relatives à un appareil de périphérie

Consultez le nom de l' AWS IoT Greengrass objet de l'appareil, l'état et la version du SSM Agent ping, etc.

- Gestion des comptes et du registre

  Gérez les comptes utilisateurs du système d'exploitation (OS) sur vos instances et le registre sur vos instances Windows.

- Contrôle de l'accès aux fonctions

  Contrôlez l'accès aux Fleet Manager fonctionnalités à l'aide de politiques AWS Identity and Access Management (IAM). Grâce à ces politiques, vous pouvez désigner les utilisateurs individuels ou les groupes de votre organisation autorisés à utiliser telle ou telle fonction de Fleet Manager, et les nœuds gérés qu'ils peuvent gérer.

Rubriques

Démarrer avec Fleet Manager

Avant de pouvoir utiliser la fonctionnalité Fleet Manager d'AWS Systems Manager pour surveiller et gérer vos nœuds gérés, vous devez suivre les étapes décrites dans les rubriques suivantes.

Rubriques

Étape 1 : créer une politique IAM avec des autorisations Fleet Manager

Pour être utiliséeFleet Manager, une fonctionnalité de AWS Systems Manager votre utilisateur ou rôle AWS Identity and Access Management (IAM) doit disposer des autorisations requises. Vous pouvez créer une politique IAM donnant accès à toutes les fonctions Fleet Manager ou modifier votre politique afin d'octroyer l'accès aux fonctions que vous sélectionnez.

Les exemples de politiques ci-dessous fournissent les autorisations requises pour toutes les fonctions Fleet Manager et les autorisations nécessaires pour des sous-ensembles de fonctions.

Pour de plus amples informations sur la création de politiques IAM, consultez Création de politiques IAM dans le guide de l'utilisateur IAM.

Rubriques

- Exemple de politique d'accès des administrateurs Fleet Manager
- Exemple de politique d'accès Fleet Manager en lecture seule

Exemple de politique d'accès des administrateurs Fleet Manager

La politique suivante fournit des autorisations pour toutes les fonctions Fleet Manager Cela signifie qu'un utilisateur peut créer et supprimer des utilisateurs et des groupes locaux, modifier l'appartenance à un groupe local et modifier les clés ou les valeurs de Windows Server registre. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"EC2",
         "Effect":"Allow",
         "Action":[
            "ec2:CreateTags",
            "ec2:DeleteTags",
            "ec2:DescribeInstances",
            "ec2:DescribeTags"
         ],
         "Resource":"*"
      },
      {
         "Sid":"General",
         "Effect":"Allow",
         "Action":[
            "ssm:AddTagsToResource",
            "ssm:DescribeInstanceAssociationsStatus",
            "ssm:DescribeInstancePatches",
            "ssm:DescribeInstancePatchStates",
            "ssm:DescribeInstanceProperties",
```

```
            "ssm:GetCommandInvocation",
            "ssm:GetServiceSetting",
            "ssm:GetInventorySchema",
            "ssm:ListComplianceItems",
            "ssm:ListInventoryEntries",
            "ssm:ListTagsForResource",
            "ssm:ListCommandInvocations",
            "ssm:ListAssociations",
            "ssm:RemoveTagsFromResource"
         ],
         "Resource":"*"
      },
      {
            "Sid":"DefaultHostManagement",
            "Effect": "Allow",
            "Action": [
                "ssm:ResetServiceSetting",
                "ssm:UpdateServiceSetting"
            ],
            "Resource": "arn:aws:ssm:region:account-id:servicesetting/ssm/managed-
instance/default-ec2-instance-management-role"
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:PassRole"
            ],
            "Resource": "arn:aws:iam::account-id:role/service-role/
AWSSystemsManagerDefaultEC2InstanceManagementRole",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": [
                        "ssm.amazonaws.com"
                    ]
                }
            }
      },
      {
         "Sid":"SendCommand",
         "Effect":"Allow",
         "Action":[
            "ssm:GetDocument",
            "ssm:SendCommand",
            "ssm:StartSession"
```

```
        ],
        "Resource":[
           "arn:aws:ec2:*:account-id:instance/*",
           "arn:aws:ssm:*:account-id:managed-instance/*",
           "arn:aws:ssm:*:account-id:document/SSM-SessionManagerRunShell",
           "arn:aws:ssm:*:*:document/AWS-PasswordReset",
           "arn:aws:ssm:*:*:document/AWSFleetManager-AddUsersToGroups",
           "arn:aws:ssm:*:*:document/AWSFleetManager-CopyFileSystemItem",
           "arn:aws:ssm:*:*:document/AWSFleetManager-CreateDirectory",
           "arn:aws:ssm:*:*:document/AWSFleetManager-CreateGroup",
           "arn:aws:ssm:*:*:document/AWSFleetManager-CreateUser",
           "arn:aws:ssm:*:*:document/AWSFleetManager-CreateUserInteractive",
           "arn:aws:ssm:*:*:document/AWSFleetManager-CreateWindowsRegistryKey",
           "arn:aws:ssm:*:*:document/AWSFleetManager-DeleteFileSystemItem",
           "arn:aws:ssm:*:*:document/AWSFleetManager-DeleteGroup",
           "arn:aws:ssm:*:*:document/AWSFleetManager-DeleteUser",
           "arn:aws:ssm:*:*:document/AWSFleetManager-DeleteWindowsRegistryKey",
           "arn:aws:ssm:*:*:document/AWSFleetManager-DeleteWindowsRegistryValue",
           "arn:aws:ssm:*:*:document/AWSFleetManager-GetDiskInformation",
           "arn:aws:ssm:*:*:document/AWSFleetManager-GetFileContent",
           "arn:aws:ssm:*:*:document/AWSFleetManager-GetFileSystemContent",
           "arn:aws:ssm:*:*:document/AWSFleetManager-GetGroups",
           "arn:aws:ssm:*:*:document/AWSFleetManager-GetPerformanceCounters",
           "arn:aws:ssm:*:*:document/AWSFleetManager-GetProcessDetails",
           "arn:aws:ssm:*:*:document/AWSFleetManager-GetUsers",
           "arn:aws:ssm:*:*:document/AWSFleetManager-GetWindowsEvents",
           "arn:aws:ssm:*:*:document/AWSFleetManager-GetWindowsRegistryContent",
           "arn:aws:ssm:*:*:document/AWSFleetManager-MountVolume",
           "arn:aws:ssm:*:*:document/AWSFleetManager-MoveFileSystemItem",
           "arn:aws:ssm:*:*:document/AWSFleetManager-RemoveUsersFromGroups",
           "arn:aws:ssm:*:*:document/AWSFleetManager-RenameFileSystemItem",
           "arn:aws:ssm:*:*:document/AWSFleetManager-SetWindowsRegistryValue",
           "arn:aws:ssm:*:*:document/AWSFleetManager-StartProcess",
           "arn:aws:ssm:*:*:document/AWSFleetManager-TerminateProcess"
        ],
        "Condition":{
           "BoolIfExists":{
              "ssm:SessionDocumentAccessCheck":"true"
           }
        }
     },
     {
        "Sid":"TerminateSession",
        "Effect":"Allow",
```

```
            "Action":[
               "ssm:TerminateSession"
            ],
            "Resource":"*",
            "Condition":{
               "StringLike":{
                  "ssm:resourceTag/aws:ssmmessages:session-id":[
                     "${aws:userid}"
                  ]
               }
            }
         },
         {
            "Sid":"KMS",
            "Effect":"Allow",
            "Action":[
               "kms:GenerateDataKey"
            ],
            "Resource":[
               "arn:aws:kms:region:account-id:key/key-name"
            ]
         }
      ]
}
```

Exemple de politique d'accès Fleet Manager en lecture seule

La politique suivante fournit des autorisations à des fonctions Fleet Manager en lecture seule. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"EC2",
         "Effect":"Allow",
         "Action":[
            "ec2:DescribeInstances",
            "ec2:DescribeTags"
         ],
         "Resource":"*"
      },
```

```
        {
            "Sid":"General",
            "Effect":"Allow",
            "Action":[
                "ssm:DescribeInstanceAssociationsStatus",
                "ssm:DescribeInstancePatches",
                "ssm:DescribeInstancePatchStates",
                "ssm:DescribeInstanceProperties",
                "ssm:GetCommandInvocation",
                "ssm:GetServiceSetting",
                "ssm:GetInventorySchema",
                "ssm:ListComplianceItems",
                "ssm:ListInventoryEntries",
                "ssm:ListTagsForResource",
                "ssm:ListCommandInvocations",
                "ssm:ListAssociations"
            ],
            "Resource":"*"
        },
        {
            "Sid":"SendCommand",
            "Effect":"Allow",
            "Action":[
                "ssm:GetDocument",
                "ssm:SendCommand",
                "ssm:StartSession"
            ],
            "Resource":[
                "arn:aws:ec2:*:account-id:instance/*",
                "arn:aws:ssm:*:account-id:managed-instance/*",
                "arn:aws:ssm:*:account-id:document/SSM-SessionManagerRunShell",
                "arn:aws:ssm:*:*:document/AWSFleetManager-GetDiskInformation",
                "arn:aws:ssm:*:*:document/AWSFleetManager-GetFileContent",
                "arn:aws:ssm:*:*:document/AWSFleetManager-GetFileSystemContent",
                "arn:aws:ssm:*:*:document/AWSFleetManager-GetGroups",
                "arn:aws:ssm:*:*:document/AWSFleetManager-GetPerformanceCounters",
                "arn:aws:ssm:*:*:document/AWSFleetManager-GetProcessDetails",
                "arn:aws:ssm:*:*:document/AWSFleetManager-GetUsers",
                "arn:aws:ssm:*:*:document/AWSFleetManager-GetWindowsEvents",
                "arn:aws:ssm:*:*:document/AWSFleetManager-GetWindowsRegistryContent"
            ],
            "Condition":{
                "BoolIfExists":{
                    "ssm:SessionDocumentAccessCheck":"true"
```

```
            }
         }
      },
      {
         "Sid":"TerminateSession",
         "Effect":"Allow",
         "Action":[
            "ssm:TerminateSession"
         ],
         "Resource":"*",
         "Condition":{
            "StringLike":{
               "ssm:resourceTag/aws:ssmmessages:session-id":[
                  "${aws:userid}"
               ]
            }
         }
      },
      {
         "Sid":"KMS",
         "Effect":"Allow",
         "Action":[
            "kms:GenerateDataKey"
         ],
         "Resource":[
            "arn:aws:kms:region:account-id:key/key-name"
         ]
      }
   ]
}
```

Étape 2 : vérifier que vos instances et appareils de périphérie peuvent être gérés par Systems Manager

Pour que les instances Amazon Elastic Compute Cloud (Amazon EC2), les appareils Core AWS IoT Greengrass ainsi que les serveurs sur site, les appareils de périphérie et les machines virtuelles (VM) puissent être surveillés et gérés à l'aide de la fonctionnalité Fleet Manager d'AWS Systems Manager, ceux-ci doivent être des nœuds gérés Systems Manager. Cela signifie que vos nœuds doivent remplir certaines conditions préalables et être configurés avec l'agent AWS Systems Manager (SSM Agent). Pour de plus amples informations, veuillez consulter Con AWS Systems Manager figuration.

Vous pouvez utiliser la capacité Quick Setup d'AWS Systems Manager pour configurer rapidement vos instances Amazon EC2 en tant qu'instances gérées sur un compte individuel. Si votre entreprise ou organisation utilise AWS Organizations, vous pouvez également configurer des instances sur plusieurs unités organisationnelles (UO) et Régions AWS. Pour de plus amples informations sur l'utilisation de Quick Setup pour configurer les instances gérées, veuillez consulter Gestion des hôtes Amazon EC2.

> **Note**
>
> Pour les machines non EC2 qui ne s'exécutent pas sur AWS, utilisez une activation hybride pour configurer la machine pour une utilisation avec Systems Manager dans un environnement hybride et multicloud. Pour obtenir des informations sur les activations hybrides, veuillez consulter AWS Systems Manager Activations hybrides.

Utilisation de l'option Fleet Manager

Vous pouvez utiliser Fleet Manager une fonctionnalité de AWS Systems Manager pour effectuer diverses tâches sur vos nœuds gérés à partir de la AWS Systems Manager console. Les rubriques suivantes décrivent les fonctions fournies par Fleet Manager.

> **Note**
>
> La seule fonction prise en charge pour les instances macOS est l'affichage du système de fichiers.

Rubriques

- Gestion des comptes utilisateur du système d'exploitation sur les nœuds gérés
- Gestion du registre Windows sur les nœuds gérés
- Accès au portail de la base de connaissances Red Hat

Utilisation de nœuds gérés

Un nœud géré est une machine configurée pour AWS Systems Manager. Vous pouvez configurer les types de machines suivants en tant que nœuds gérés :

- Instances Amazon Elastic Compute Cloud (Amazon EC2)
- Serveurs sur votre propre site (serveurs sur site)
- AWS IoT Greengrass appareils principaux
- AWS IoT et appareils non AWS périphériques
- Machines virtuelles (VM), y compris les VM dans d'autres environnements cloud

> **Note**
>
> Dans la console Systems Manager, toute machine dotée du préfixe « mi- » est configurée en tant que nœud géré à l'aide d'une activation hybride. Les appareils de périphérie affichent leur nom d'objet AWS IoT .

AWS Systems Manager propose un niveau d'instances standard et un niveau d'instances avancées. Les deux prennent en charge les nœuds gérés dans votre environnement hybride et multicloud. Le niveau d'instances standard vous permet d'enregistrer un maximum de 1 000 machines par machine. Compte AWS Région AWS Si vous avez besoin d'enregistrer plus de 1 000 machines dans un seul compte et une seule région, utilisez le niveau d'instances avancées. Le niveau d'instances avancées vous permet de créer autant de nœuds gérés que vous le souhaitez. Tous les nœuds gérés configurés pour Systems Manager sont facturés sur une pay-per-use base. Pour plus d'informations sur l'activation des instances avancées, consultez Activation du niveau d'instances avancées. Pour plus d'informations sur la tarification, consultez Tarification AWS Systems Manager.

> **Note**
>
> - Les instances avancées vous permettent également de vous connecter à vos nœuds non EC2 dans un environnement hybride et multicloud en utilisant. AWS Systems Manager Session Manager Session Managerfournit un accès shell interactif à vos instances. Pour plus d'informations, consultez AWS Systems Manager Session Manager.
> - 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).
> - Pour corriger les applications publiées par Microsoft sur des instances de machines virtuelles (VM) sur site, activez le niveau d'instances avancées. L'utilisation du niveau d'instance avancé est facturée. La correction d'applications publiées par Microsoft sur des instances Amazon Elastic Compute Cloud (Amazon EC2) n'induit aucuns frais supplémentaires. Pour plus d'informations, consultez À propos de la correction d'applications publiées par Microsoft sur Windows Server.

Affichage des nœuds gérés

Si vos nœuds gérés ne sont pas répertoriés dans la console, procédez comme suit :

1. Vérifiez que la console est ouverte Région AWS là où vous avez créé vos nœuds gérés. Vous pouvez changer de région à l'aide de la liste figurant dans la partie supérieure droite de la console.
2. Vérifiez que les étapes de configuration de vos nœuds gérés respectent les exigences de Systems Manager. Pour plus d'informations, veuillez consulter Con AWS Systems Manager figuration.
3. Pour les machines non EC2, vérifiez que vous avez terminé le processus d'activation hybride. Pour plus d'informations, consultez Utilisation de Systems Manager dans des environnements hybrides et multicloud.

> **Note**
>
> Notez les informations suivantes.
>
> - La console Fleet Manager n'affiche pas les nœuds Amazon EC2 qui ont été résiliés.
> - Systems Manager doit disposer de références précises en termes de date et d'heure pour pouvoir effectuer des opérations sur vos machines. Si la date et l'heure ne sont pas correctement définies sur vos nœuds gérés, les machines peuvent ne pas correspondre

à la date de signature de vos demandes d'API. Pour plus d'informations, consultez Cas d'utilisation et bonnes pratiques.

- Lorsque vous créez ou modifiez des balises, une heure peut s'écouler avant que le système n'affiche les modifications dans le filtre de la table.
- Une fois que le statut d'un nœud géré a été `Connection Lost` pendant au moins 30 jours, il est possible que le nœud ne soit plus répertorié dans la console Fleet Manager. Pour le rétablir dans la liste, le problème à l'origine de la perte de connexion doit être résolu. Pour obtenir des conseils de dépannage, veuillez consulter Résolution des problèmes de disponibilité des nœuds gérés.

Vérification de la prise en charge de Systems Manager sur un nœud géré

AWS Config fournit des règles AWS gérées, qui sont des règles prédéfinies et personnalisables AWS Config utilisées pour évaluer si vos configurations de AWS ressources sont conformes aux meilleures pratiques courantes. AWS Config Les règles gérées incluent la règle ec2-instance-managed-by-systems-manager. Cette règle vérifie si les instances Amazon EC2 de votre compte sont gérées par Systems Manager. Pour en savoir plus, consultez Règles gérées AWS Config.

Renforcement de la sécurité sur les nœuds gérés

Pour plus d'informations sur le renforcement de votre posture de sécurité vis-à-vis des commandes de niveau racine non autorisées sur vos nœuds gérés, consultez Limitation de l'accès aux commandes de niveau racine via l'SSM Agent.

Annulation de l'enregistrement des nœuds gérés

Vous pouvez à tout moment annuler l'enregistrement des nœuds gérés. Par exemple, si vous gérez plusieurs nœuds dotés du même rôle AWS Identity and Access Management (IAM) et que vous remarquez un quelconque comportement malveillant, vous pouvez annuler l'enregistrement d'un nombre illimité de machines à tout moment. Pour obtenir des informations sur l'annulation de l'enregistrement des nœuds gérés, consultez Annulation de l'enregistrement de nœuds gérés dans un environnement hybride et multicloud.

Rubriques

- Configuration des niveaux d'instance
- Réinitialisation des mots de passe sur les nœuds gérés
- Annulation de l'enregistrement de nœuds gérés dans un environnement hybride et multicloud

Configuration des niveaux d'instance

Cette rubrique décrit les scénarios dans lesquels vous devez activer le niveau d'instances avancées.

AWS Systems Manager propose un niveau d'instances standard et un niveau d'instances avancées pour les machines non EC2 dans un environnement hybride et multicloud.

Vous pouvez enregistrer jusqu'à 1 000 nœuds hybrides standard par compte et sans Région AWS frais supplémentaires. Toutefois, pour enregistrer plus de 1 000 nœuds hybrides, vous avez besoin d'activer le niveau d'instances avancées. L'utilisation du niveau d'instance avancé est facturée. Pour plus d'informations, consultez Tarification d'AWS Systems Manager.

Même avec moins de 1 000 nœuds activés par un système hybride enregistrés, deux autres scénarios requièrent le niveau d'instances avancées :

- Vous souhaitez utiliser Session Manager pour vous connecter à des nœuds non EC2.
- Vous souhaitez appliquer des correctifs aux applications (et non aux systèmes d'exploitation) publiées par Microsoft sur des nœuds non EC2.

> **Note**
>
> Les opérations d'application de correctifs publiées par Microsoft sur des instances Amazon EC2 sont gratuites.

Scénarios détaillés relatifs au niveau d'instances avancées

Les informations suivantes fournissent des détails relatifs aux trois scénarios pour lesquels vous devez activer le niveau d'instances avancées.

Scénario 1 : vous souhaitez enregistrer plus de 1 000 nœuds activés par un système hybride

En utilisant le niveau d'instances standard, vous pouvez enregistrer 1 000 nœuds non EC2 au maximum dans un environnement hybride et multicloud par Région AWS sur un compte spécifique sans frais supplémentaires. Si vous avez besoin d'enregistrer plus de 1 000 nœuds non EC2 dans une région, vous devez utiliser le niveau d'instances avancées. Vous pouvez ensuite activer autant de machines d'un environnement hybride et multicloud que vous souhaitez. Les frais pour le niveau d'instances avancées sont basés sur, le nombre de nœuds avancés activés en tant que nœuds gérés par Systems Manager et le nombre d'heures d'exécution de ces nœuds.

Tous les nœuds gérés par Systems Manager qui utilisent le processus d'activation décrit dans Créer une activation hybride pour enregistrer des nœuds auprès de Systems Manager sont alors soumis à des frais si vous dépassez les 1 000 nœuds locaux dans une région sur un compte spécifique.

> **Note**
>
> Vous pouvez également activer les instances Amazon Elastic Compute Cloud (Amazon EC2) existantes à l'aide des activations hybrides Systems Manager et les utiliser en tant qu'instances non EC2, comme pour les tests. Ces nœuds sont également considérés comme des nœuds hybrides. Ce scénario n'est pas courant.

Scénario 2 : application des correctifs à des applications publiées par Microsoft sur des nœuds activés par des hybrides

Le niveau d'instances avancées est également requis si vous souhaitez appliquer des correctifs à des applications publiées par Microsoft sur des nœuds non EC2 dans un environnement hybride et multicloud. L'activation du niveau d'instances avancées pour appliquer des correctifs aux applications Microsoft sur des nœuds non EC2, des frais seront alors facturés pour tous les nœuds sur site, même si vous en avez moins de 1 000.

La correction d'applications publiées par Microsoft sur des instances Amazon Elastic Compute Cloud (Amazon EC2) n'induit aucuns frais supplémentaires. Pour plus d'informations, consultez À propos de la correction d'applications publiées par Microsoft sur Windows Server.

Scénario 3 : connexion à des nœuds activés par des hybrides à l'aide de Session Manager

Session Manager fournit un accès shell interactif à vos instances. Pour vous connecter à des nœuds gérés activés par un système hybride à l'aide de Session Manager, activez le niveau d'instances avancées. Des frais sont ensuite facturés pour tous les nœuds activés par un système hybride, même si vous en avez moins de 1 000.

Résumé : quand ai-je besoin du niveau d'instances avancées ?

Utilisez le tableau suivant pour savoir quand utiliser le niveau d'instances avancées et quels scénarios impliquent des frais supplémentaires.

Scénario	Le niveau d'instances avancées est requis ?	Des frais supplémentaires seront facturés.
Le nombre de nœuds activés par des hybrides dans ma région sur un compte spécifique est supérieur à 1 000.	Oui	Oui
Je souhaite utiliser Patch Manager pour appliquer des correctifs aux applications publiées par Microsoft sur un nombre illimité de nœuds activés par des hybrides, même moins de 1 000.	Oui	Oui
Je souhaite utiliser Session Manager pour vous connecter à n'importe quel nombre de nœuds activés par des hybrides, même à moins de 1 000.	Oui	Oui
1. Le nombre de nœuds activés par des hybrides dans ma région sur un compte spécifique est supérieur à 1 000. 2. Je n'applique aucun correctif aux applications Microsoft sur des nœuds activés par des hybrides ; et 3. Je ne me connecte à aucun nœud activé par un hybride en utilisant Session Manager.	Non	Non

Rubriques

- Activation du niveau d'instances avancées
- Revenir du niveau des instances avancées au niveau des instances standard

Activation du niveau d'instances avancées

AWS Systems Manager propose un niveau d'instances standard et un niveau d'instances avancées pour les machines non EC2 dans un environnement hybride et multicloud. Le niveau d'instances standard vous permet d'enregistrer un maximum de 1 000 machines sur site par Compte AWS et par Région AWS. Le niveau d'instances avancées est également nécessaire pour utiliser Patch Manager pour appliquer des correctifs à des applications publiées par Microsoft sur des nœuds non EC2 et pour se connecter à des nœuds non EC2 à l'aide deSession Manager. Pour plus d'informations, consultez Configuration des niveaux d'instance.

Cette section décrit comment configurer votre environnement hybride et multicloud 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 disponibles sur un per-use-basis. Pour plus d'informations, consultez Tarification AWS Systems Manager.

Configuration des autorisations pour activer le niveau des instances avancées.

Vérifiez que vous êtes autorisé AWS Identity and Access Management (IAM) à modifier votre environnement du niveau des instances standard au niveau des instances avancées. La politique IAM `AdministratorAccess` doit être attachée à votre utilisateur , groupe ou rôle, ou vous devez disposer d'une autorisation pour modifier le paramètre de service de niveau d'activation Systems manager. Le paramètre de niveau d'activation utilise les opérations d'API suivantes :

- GetServiceSetting
- UpdateServiceSetting
- ResetServiceSetting

Utilisez la procédure suivante pour ajouter une politique IAM en ligne à un compte utilisateur. Cette politique permet à un utilisateur d'afficher le paramètre actuel de niveau d'instance gérée. Cette

politique permet également à l'utilisateur de modifier ou de réinitialiser le paramètre actuel dans les valeurs spécifiées Compte AWS et Région AWS.

1. Connectez-vous à la console IAM AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation, sélectionnez Users.
3. Dans la liste, sélectionnez le nom de l'utilisateur auquel intégrer une politique.
4. Choisissez l'onglet Permissions (Autorisations).
5. Sur le côté droit de la page, sous Permission policies (Politiques d'autorisation), sélectionnez Add inline policy (Ajouter une politique en ligne).
6. Sélectionnez 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. Sélectionnez Examiner une politique.
9. Sur la page Examiner une politique, dans le champ Nom, saisissez un nom pour la politique en ligne. Par exemple : **`Managed-Instances-Tier`**.

10. Sélectionnez Créer une politique.

Les administrateurs peuvent spécifier une autorisation en lecture seule en affectant la politique en ligne suivante à 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 de politiques IAM, consultez Création de politiques IAM dans le guide de l'utilisateur IAM.

Activation du niveau d'instances avancées (console)

La procédure suivante explique comment utiliser la console Systems Manager pour modifier tous les nœuds non EC2 ajoutés à l'aide de l'activation des instances gérées, dans le niveau spécifié Compte AWS et Région AWS pour utiliser le niveau des instances avancées.

Avant de commencer

Vérifiez que la console est ouverte Région AWS là 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.

Vérifiez que vous avez satisfait la configuration requise pour vos instances Amazon Elastic Compute Cloud (Amazon EC2) et vos machines non EC2 dans un environnement hybride et multicloud. Pour plus d'informations, veuillez consulter Con AWS Systems Manager figuration.

> **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 AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Choisissez Paramètres, Modifier les paramètres du niveau d'instance.
4. Vérifiez les informations contenues dans la boîte de dialogue sur la modification des paramètres de compte, puis continuez.
5. Si vous approuvez, choisissez l'option d'acceptation, puis choisissez Modifier le paramètre.

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, consultez Revenir du niveau des instances avancées au niveau des instances standard.

Activation du niveau d'instances avancées (AWS CLI)

La procédure suivante explique comment utiliser le AWS Command Line Interface pour modifier tous les serveurs locaux et machines virtuelles ajoutés à l'aide de l'activation des instances gérées, dans le niveau spécifié Compte AWS et Région AWS pour utiliser le niveau des instances avancées.

> **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 du AWS CLI

1. Ouvrez le AWS CLI et exécutez la commande suivante. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   Linux & macOS

   ```
   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 nœuds gérés dans les versions actuelles Compte AWS et Région AWS.

   Linux & macOS

   ```
   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 explique comment utiliser le AWS Tools for Windows PowerShell pour modifier tous les serveurs locaux et machines virtuelles ajoutés à l'aide de l'activation des instances gérées, dans le niveau spécifié Compte AWS et Région AWS pour utiliser le niveau des instances avancées.

> **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 de PowerShell

1. Ouvrez AWS Tools for Windows PowerShell et exécutez la commande suivante. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

```
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 nœuds gérés dans les versions actuelles Compte AWS et Région AWS.

```
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
```

Plusieurs minutes peuvent s'écouler avant que les nœuds passent tous du niveau d'instances standard au niveau d'instances avancées.

> **Note**
>
> Pour de plus amples informations sur le retour au niveau des instances standard, consultez Revenir du niveau des instances avancées au niveau des instances standard.

Revenir du niveau des instances avancées au niveau des instances standard

Cette section explique comment rebasculer des nœuds activés par un système hybride du niveau d'instances avancées vers le niveau d'instances standard. Cette configuration s'applique à tous les nœuds activés de manière hybride en un Compte AWS et un. Région AWS

Avant de commencer

Consultez les détails importants suivants.

Note

- Vous ne pouvez pas revenir au niveau d'instances standard si vous exécutez plus de 1 000 nœuds activés par un système hybride dans le compte et la région. Vous devez d'abord annuler l'enregistrement des nœuds jusqu'à ce que vous en ayez 1 000 ou moins. Cela concerne également les instances Amazon Elastic Compute Cloud (Amazon EC2) qui utilisent une activation hybride Systems Manager (ce qui n'est pas courant). Pour plus d'informations, consultez Annulation de l'enregistrement de nœuds gérés dans un environnement hybride et multicloud.
- Une fois que vous serez revenu, vous ne pourrez plus utiliser Session Manager la fonctionnalité permettant d'accéder de AWS Systems Manager manière interactive à vos nœuds activés par des systèmes hybrides.
- Une fois que vous serez rétabli, vous ne pourrez plus utiliser Patch Manager une fonctionnalité permettant de AWS Systems Manager patcher les applications publiées par Microsoft sur des nœuds activés par des systèmes hybrides.
- Le processus de retour en arrière de tous les nœuds activés par un système hybride au niveau d'instances standard peut prendre 30 minutes ou plus.

Cette section décrit comment rétablir tous les nœuds activés par des hybrides dans un Compte AWS et Région AWS depuis le niveau d'instances avancées vers le niveau d'instances standard.

Revenir au niveau des instances standard (console)

La procédure suivante explique comment utiliser la console Systems Manager pour modifier tous les nœuds activés en mode hybride dans votre environnement hybride et multicloud afin d'utiliser le niveau d'instances standard dans le niveau spécifié et. Compte AWS Région AWS

Pour revenir au niveau des instances standard (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Sélectionnez le menu déroulant Account settings (Paramètres du compte) et sélectionnez Instance tier settings (Paramètres du niveau d'instances).
4. Sélectionnez Modifier les paramètres de compte.

5. Vérifiez les informations contenues dans la fenêtre sur la modification des paramètres de compte puis, si vous approuvez, sélectionnez l'option pour accepter et continuer.

Revenir au niveau des instances standard (AWS CLI)

La procédure suivante vous montre comment utiliser le AWS Command Line Interface pour modifier tous les nœuds activés par hybride dans votre environnement hybride et multicloud afin d'utiliser le niveau d'instances standard dans le niveau spécifié et. Compte AWS Région AWS

Pour revenir au niveau des instances standard à l'aide du AWS CLI

1. Ouvrez le AWS CLI et exécutez la commande suivante. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   Linux & macOS

   ```
   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 les versions actuelles Compte AWS et Région AWS.

   Linux & macOS

   ```
   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": "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 explique comment modifier les nœuds activés AWS Tools for Windows PowerShell en mode hybride dans votre environnement hybride et multicloud afin d'utiliser le niveau d'instances standard dans le niveau spécifié et. Compte AWS Région AWS

Pour revenir au niveau des instances standard à l'aide de PowerShell

1. Ouvrez AWS Tools for 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 les versions actuelles Compte AWS et Région AWS.

```
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 des mots de passe sur les nœuds gérés

Vous pouvez réinitialiser le mot de passe de n'importe quel utilisateur d'un nœud géré. Cela inclut les instances Amazon Elastic Compute Cloud (Amazon EC2), les appareils principaux AWS IoT Greengrass , ainsi que les serveurs sur site, les appareils périphériques et les machines virtuelles (VM) gérés par. AWS Systems Manager La fonctionnalité de réinitialisation du mot de passe repose sur Session Manager une fonctionnalité de AWS Systems Manager. Vous pouvez utiliser cette fonctionnalité pour vous connecter aux nœuds gérés sans avoir à ouvrir de ports entrants, à maintenir des hôtes bastions ou à gérer des clés SSH.

La réinitialisation de mot de passe peut s'avérer utile lorsqu'un utilisateur a oublié un mot de passe, ou lorsque vous souhaitez modifier un mot de passe rapidement sans établir de connexion SSH ou RDP avec un nœud géré.

Prérequis

Avant de pouvoir réinitialiser le mot de passe d'un nœud géré, les conditions suivantes doivent être remplies :

- Le nœud géré sur lequel vous souhaitez modifier le mot de passe doit être un nœud géré Systems Manager. De plus, SSM Agent 2.3.668.0 (ou version ultérieure) doit être installé sur le nœud géré.

Pour plus d'informations sur l'installation ou la mise à jour de SSM Agent, consultez Utilisation de l'option SSM Agent.

- La fonctionnalité de réinitialisation de mot de passe utilise la configuration Session Manager définie pour permettre à votre compte de se connecter au nœud géré. Par conséquent, les prérequis pour l'utilisation de Session Manager doivent avoir été satisfaits pour votre compte dans la Région AWS actuelle. Pour plus d'informations, consultez Configuration de Session Manager.

  > **Note**
  >
  > La prise en charge de Session Manager pour les nœuds sur site est assurée pour le niveau d'instances avancées uniquement. Pour plus d'informations, consultez Activation du niveau d'instances avancées.

- L' AWS utilisateur qui modifie le mot de passe doit avoir l'`ssm:SendCommand`autorisation d'accéder au nœud géré. Pour plus d'informations, consultez Restriction de l'accès Run Command en fonction des balises.

Restriction de l'accès

La capacité d'un utilisateur à réinitialiser les mots de passe peut être limitée à des nœuds gérés spécifiques. Pour ce faire, utilisez des politiques basées sur une identité pour l'opération Session Manager `ssm:StartSession` avec le document SSM `AWS-PasswordReset`. Pour de plus amples informations, consultez Contrôler les accès de session utilisateur aux instances.

Chiffrement de données

Activez AWS Key Management Service (AWS KMS) le chiffrement complet Session Manager des données afin d'utiliser l'option de réinitialisation du mot de passe pour les nœuds gérés. Pour plus d'informations, consultez Activer le chiffrement des données de session par clé KMS (console).

Réinitialisation d'un mot de passe sur un nœud géré

Vous pouvez réinitialiser un mot de passe sur un nœud géré par Systems Manager à l'aide de la Fleet Managerconsole Systems Manager ou du AWS Command Line Interface (AWS CLI).

Pour modifier le mot de passe sur un nœud géré (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.

2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Cliquez sur le bouton situé en regard du nœud qui a besoin d'un nouveau mot de passe.
4. Choisissez Actions de l'instance, Réinitialiser le mot de passe.
5. Pour User name (Nom d'utilisateur), saisissez le nom de l'utilisateur pour lequel vous modifiez le mot de passe. Il peut s'agir de n'importe quel utilisateur qui dispose d'un compte sur le nœud.
6. Sélectionnez Submit (Envoyer).
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 SSM Agent disponible sur le nœud géré ne prend pas en charge la réinitialisation des mots de passe, vous êtes invité à installer une version prise en charge à l'aide de la fonctionnalité Run Command d' AWS Systems Manager.

Pour réinitialiser le mot de passe sur un nœud géré (AWS CLI)

1. Pour réinitialiser le mot de passe d'un utilisateur sur un nœud géré, exécutez la commande suivante. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

> **Note**
>
> Pour utiliser le AWS CLI pour réinitialiser un mot de passe, le Session Manager plugin doit être installé sur votre machine locale. Pour plus d'informations, veuillez consulter Installez le Session Manager plugin pour AWS CLI.

Linux & macOS

```
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"
```

2. Suivez les instructions dans la fenêtre de commande Enter new password (Entrer un nouveau mot) pour spécifier le nouveau mot de passe.

Résolution des problèmes de réinitialisation de mot de passe sur les nœuds gérés

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. Pour les autres problèmes, utilisez les informations suivantes pour vous aider à résoudre les problèmes de réinitialisation de mot de passe.

Rubriques

- Nœud géré non disponible
- SSM Agentpas up-to-date (console)
- Les options de réinitialisation du mot de passe ne sont pas fournies (AWS CLI)
- Pas d'autorisation pour exécuter ssm:SendCommand
- Message d'erreur Session Manager

Nœud géré non disponible

Problème : vous souhaitez réinitialiser le mot de passe d'un nœud géré sur la page Managed instances (Instances gérées) de la console, mais le nœud ne figure pas dans la liste.

- Solution : le nœud géré auquel vous souhaitez vous connecter n'est peut-être pas configuré pour Systems Manager. Pour utiliser une instance EC2 avec Systems Manager, un profil d'instance AWS Identity and Access Management (IAM) autorisant Systems Manager à effectuer des actions sur vos instances doit être attaché à l'instance. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager.

  Pour utiliser une machine non EC2 avec Systems Manager, créez une fonction du service IAM qui accorde à Systems Manager l'autorisation d'effectuer des actions sur vos nœuds gérés. Pour

plus d'informations, voir Créer le rôle de service IAM requis pour Systems Manager dans les environnements hybrides et multicloud. (le Session Manager support pour les serveurs locaux et les machines virtuelles n'est fourni que pour le niveau des instances avancées. Pour plus d'informations, voirActivation du niveau d'instances avancées.)

SSM Agentpas up-to-date (console)

Problème : un message indique que la version de SSM Agent ne prend pas en charge la fonctionnalité de réinitialisation de mot de passe.

- Solution : la version 2.3 668.0 ou suivante de SSM Agent est requise pour effectuer des réinitialisations de mot de passe. Dans la console, sélectionnez Update SSM Agent (Mettre à jour l'agent SSM) pour procéder à la mise à jour de l'agent sur le nœud géré.

  Une nouvelle version de SSM Agent est lancée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Le fait de ne pas utiliser la dernière version de l'agent peut empêcher votre nœud géré d'utiliser diverses capacités et fonctionnalités de Systems Manager. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir SSM Agent à jour sur vos machines. Pour plus d'informations, consultez Automatisation des mises à jour de l'SSM Agent. Abonnez-vous à la page des notes de SSM Agent publication GitHub pour recevoir des notifications concernant les SSM Agent mises à jour.

Les options de réinitialisation du mot de passe ne sont pas fournies (AWS CLI)

Problème : vous vous connectez correctement à un nœud géré 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 de SSM Agent sur le nœud géré ne l'est pas up-to-date. La version 2.3.668.0 ou suivante est requise pour effectuer des réinitialisations de mot de passe.

  Une nouvelle version de SSM Agent est lancée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Le fait de ne pas utiliser la dernière version de l'agent peut empêcher votre nœud géré d'utiliser diverses capacités et fonctionnalités de Systems Manager. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir SSM Agent à jour sur vos machines. Pour plus d'informations, consultez Automatisation des mises à jour de l'SSM

Agent. Abonnez-vous à la page des notes de SSM Agent publication GitHub pour recevoir des notifications concernant les SSM Agent mises à jour.

Pas d'autorisation pour exécuter **`ssm:SendCommand`**

Problème : vous essayez de vous connecter à un nœud géré pour changer le mot de passe mais vous recevez un message d'erreur indiquant que vous n'êtes pas autorisé à exécuter `ssm:SendCommand` sur le nœud géré.

- Solution : Votre politique IAM doit inclure l'autorisation d'exécuter la commande `ssm:SendCommand`. Pour plus d'informations, consultez Restriction de l'accès Run Command en fonction des balises.

Message d'erreur Session Manager

Problème : vous recevez un message d'erreur relatif à Session Manager.

- Solution : la prise en charge de la réinitialisation de mot de passe nécessite que Session Manager soit configuré correctement. Pour plus d'informations, consultez Configuration de Session Manager et Résolution des problèmes de Session Manager.

Annulation de l'enregistrement de nœuds gérés dans un environnement hybride et multicloud

Si vous ne souhaitez plus gérer un serveur sur site, un périphérique périphérique ou une machine virtuelle (VM) en utilisant AWS Systems Manager, vous pouvez le désenregistrer. Le désenregistrement d'un nœud activé par hybride le supprime de la liste des nœuds gérés dans Systems Manager. AWS Systems Manager L'agent (SSM Agent) exécuté sur le nœud activé par l'hybride ne pourra pas actualiser son jeton d'autorisation car il n'est plus enregistré. SSM Agenthiberne et réduit sa fréquence de ping vers Systems Manager dans le cloud à une fois par heure.

Vous pouvez ré-enregistrer un serveur sur site, un appareil de périphérie ou une machine virtuelle à tout moment. Systems Manager stocke l'historique des commandes d'un nœud géré dont l'enregistrement a été annulé pendant 30 jours.

La procédure suivante décrit comment annuler l'enregistrement d'un nœud activé par un système hybride à l'aide de la console Systems Manager. Pour de plus amples informations sur la façon

de procéder à l'aide de la AWS Command Line Interface , veuillez consulter deregister-managed-instance.

Pour annuler l'enregistrement d'un nœud activé par un système hybride (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Cochez la case à côté du nœud géré que vous souhaitez désenregistrer.
4. Choisissez Actions du nœud, Outils, Désenregistrer ce nœud géré.
5. Consultez les informations de la boîte de dialogue Désenregistrer ce nœud géré. Si vous êtes d'accord, choisissez Désenregistrer.

Utilisation du paramètre de configuration de gestion d'hôte par défaut

Le paramètre de configuration de gestion d'hôte par défaut AWS Systems Manager permet de gérer automatiquement vos instances Amazon EC2 en tant qu'instances gérées. Une instance gérée est une instance EC2 configurée pour une utilisation avec Systems Manager.

Les avantages de la gestion de vos instances avec Systems Manager sont les suivants :

- Connectez-vous à vos instances EC2 en toute sécurité à l'aide de Session Manager.
- Effectuez des analyses de correctifs automatisées à l'aide de Patch Manager.
- Consultez les informations détaillées sur vos instances à l'aide de Systems Manager Inventory.
- Suivez et gérez les instances à l'aide de Fleet Manager.
- Maintenez l'SSM Agent à jour automatiquement.

Fleet Manager, Inventory, Patch Manager, et Session Manager sont des fonctionnalités de Systems Manager.

La configuration de gestion d'hôte par défaut permet de gérer les instances EC2 sans avoir à créer manuellement un profil d'instance AWS Identity and Access Management (IAM). Au lieu de cela, la configuration de gestion d'hôte par défaut crée et applique un rôle IAM par défaut afin de garantir que Systems Manager dispose des autorisations nécessaires pour gérer toutes les instances dans le Compte AWS et à l' Région AWS endroit où il est activé.

Si les autorisations fournies ne sont pas suffisantes pour votre cas d'utilisation, vous pouvez également ajouter des politiques au rôle IAM par défaut créé par la configuration de gestion des hôtes par défaut. Sinon, si vous n'avez pas besoin d'autorisations pour toutes les fonctionnalités fournies par le rôle IAM par défaut, vous pouvez créer vos propres rôles et politiques personnalisés. Toutes les modifications apportées au rôle IAM que vous choisissez pour la configuration de gestion des hôtes par défaut s'appliquent à toutes les instances Amazon EC2 gérées dans la région et le compte.

Pour plus d'informations sur la politique utilisée par la Configuration de gestion des hôtes par défaut, consultez AWS stratégie gérée : politique InstanceDefault AmazonSSMManageDec2.

Implémentation d'un accès sur la base du moindre privilège

Les procédures de la présente rubrique sont destinées à être exécutées uniquement par les administrateurs. Par conséquent, nous recommandons d'implémenter l'accès avec le moindre privilège afin d'empêcher les utilisateurs non administrateurs de configurer ou de modifier la configuration de gestion des hôtes par défaut. Pour consulter des exemples de politiques qui limitent l'accès à la configuration de gestion des hôtes par défaut, voir Exemples de politique du moindre privilège pour la configuration de gestion des hôtes par défaut plus loin dans cette rubrique.

> **Important**
>
> Les informations d'enregistrement des instances enregistrées à l'aide de la configuration de gestion d'hôte par défaut sont stockées localement dans les `C:\ProgramData\Amazon` répertoires `var/lib/amazon/ssm` or. La suppression de ces répertoires ou de leurs fichiers empêchera l'instance d'acquérir les informations d'identification nécessaires pour se connecter à Systems Manager à l'aide de la configuration de gestion des hôtes par défaut. Dans ces cas, vous devez utiliser un profil d'instance IAM pour fournir les autorisations requises à votre instance ou recréer l'instance.

Rubriques

Prérequis

Pour utiliser la configuration de gestion d'hôte par défaut dans le paramètre Région AWS et à l' Compte AWS endroit où vous activez le paramètre, les conditions suivantes doivent être remplies.

- Une instance à gérer doit utiliser le service des métadonnées d'instance version 2 (IMDSv2).

  La configuration de gestion des hôtes par défaut ne prend pas en charge du Service des métadonnées d'instance Version 1. Pour plus d'informations sur la transition vers IMDSv2, consultez la section Transition vers l'utilisation du service de métadonnées d'instance version 2 dans le guide de l'utilisateur Amazon EC2
- SSM Agent version 3.2.582.0 ou une version ultérieure doit être installé sur l'instance à gérer.

  Pour plus d'informations sur la vérification de la version de l'SSM Agent installée sur votre instance, consultez Vérification du numéro de version de l'SSM Agent.

  Pour plus d'informations sur la mise à jour de l'SSM Agent, veuillez consulter la rubrique Mise à jour automatique de l'SSM Agent.
- En tant qu'administrateur effectuant les tâches décrites dans cette rubrique, vous devez disposer des autorisations nécessaires pour les GetService opérations d'API ResetServiceSetting, UpdateServiceSetting et Setting. En outre, vous devez disposer des autorisations nécessaires pour l'autorisation `iam:PassRole` du rôle IAM `AWSSystemsManagerDefaultEC2InstanceManagementRole`. Voici un exemple de politique fournissant ces autorisations. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetServiceSetting",
                "ssm:ResetServiceSetting",
                "ssm:UpdateServiceSetting"
            ],
            "Resource": "arn:aws:ssm:region:account-id:servicesetting/ssm/managed-
instance/default-ec2-instance-management-role"
        },
        {
```

```
            "Effect": "Allow",
            "Action": [
                "iam:PassRole"
            ],
            "Resource": "arn:aws:iam::account-id:role/service-role/
AWSSystemsManagerDefaultEC2InstanceManagementRole",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": [
                        "ssm.amazonaws.com"
                    ]
                }
            }
        }
    ]
}
```

- Si un profil d'instance IAM est déjà attaché à une instance EC2 à gérer à l'aide de Systems Manager, vous devez supprimer toutes les autorisations qui permettent l'opération `ssm:UpdateInstanceInformation`. Le SSM Agent tente d'utiliser les autorisations du profil d'instance avant d'utiliser les autorisations de la Configuration de gestion des hôtes par défaut. Si vous autorisez l'opération `ssm:UpdateInstanceInformation` dans votre propre profil d'instance IAM, l'instance n'utilisera pas les autorisations de configuration de gestion des hôtes par défaut.

Activation du paramètre de configuration de gestion d'hôte par défaut

Vous pouvez activer la configuration de gestion d'hôte par défaut depuis la Fleet Manager console ou en utilisant le AWS Command Line Interface ou AWS Tools for Windows PowerShell.

Vous devez activer la configuration de gestion d'hôte par défaut une par une dans chaque région où vous souhaitez que vos instances Amazon EC2 soient gérées selon ce paramètre.

Après avoir activé la configuration de gestion d'hôte par défaut, vos instances peuvent prendre jusqu'à 30 minutes pour utiliser les informations d'identification du rôle que vous avez choisi à l'étape 5 de la procédure suivante.

Pour activer la configuration de gestion des hôtes par défaut (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.

2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Choisissez Gestion de compte, Configuration de la gestion des hôtes par défaut.
4. Activez l'option Activer la configuration de gestion des hôtes par défaut.
5. Choisissez le rôle AWS Identity and Access Management (IAM) utilisé pour activer les fonctionnalités de Systems Manager pour vos instances. Nous vous recommandons d'utiliser le rôle par défaut dans Configuration de gestion des hôtes par défaut. Il contient l'ensemble des autorisations minimum pour gérer vos instances Amazon EC2 à l'aide de Systems Manager. Si vous préférez utiliser un rôle personnalisé, la politique de confiance du rôle doit autoriser Systems Manager en tant qu'entité de confiance.
6. Choisissez Configurer pour terminer la configuration.

Pour activer la configuration de gestion des hôtes par défaut (ligne de commande)

1. Créez un fichier JSON sur votre machine locale, contenant la politique de relation de confiance.

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

2. Ouvrez le AWS CLI ou les outils pour Windows PowerShell et exécutez l'une des commandes suivantes, en fonction du type de système d'exploitation de votre ordinateur local, pour créer un rôle de service dans votre compte. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   Linux & macOS

```
aws iam create-role \
--role-name AWSSystemsManagerDefaultEC2InstanceManagementRole \
--path /service-role/ \
```

```
--assume-role-policy-document file://trust-policy.json
```

Windows

```
aws iam create-role ^
--role-name AWSSystemsManagerDefaultEC2InstanceManagementRole ^
--path /service-role/ ^
--assume-role-policy-document file://trust-policy.json
```

PowerShell

```
New-IAMRole `
-RoleName "AWSSystemsManagerDefaultEC2InstanceManagementRole" `
-Path "/service-role/" `
-AssumeRolePolicyDocument "file://trust-policy.json"
```

3. Exécutez la commande suivante pour attacher la politique gérée `AmazonSSMManagedEC2InstanceDefaultPolicy` au rôle que vous venez de créer. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   Linux & macOS

   ```
   aws iam attach-role-policy \
   --policy-arn arn:aws:iam::aws:policy/AmazonSSMManagedEC2InstanceDefaultPolicy \
   --role-name AWSSystemsManagerDefaultEC2InstanceManagementRole
   ```

   Windows

   ```
   aws iam attach-role-policy ^
   --policy-arn arn:aws:iam::aws:policy/AmazonSSMManagedEC2InstanceDefaultPolicy ^
   --role-name AWSSystemsManagerDefaultEC2InstanceManagementRole
   ```

   PowerShell

   ```
   Register-IAMRolePolicy `
   -PolicyArn "arn:aws:iam::aws:policy/AmazonSSMManagedEC2InstanceDefaultPolicy" `
   -RoleName "AWSSystemsManagerDefaultEC2InstanceManagementRole"
   ```

4. Ouvrez le AWS CLI ou les outils pour Windows PowerShell et exécutez la commande suivante. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   Linux & macOS

   ```
   aws ssm update-service-setting \
   --setting-id arn:aws:ssm:region:account-id:servicesetting/ssm/managed-instance/
   default-ec2-instance-management-role \
   --setting-value service-role/AWSSystemsManagerDefaultEC2InstanceManagementRole
   ```

   Windows

   ```
   aws ssm update-service-setting ^
   --setting-id arn:aws:ssm:region:account-id:servicesetting/ssm/managed-instance/
   default-ec2-instance-management-role ^
   --setting-value service-role/AWSSystemsManagerDefaultEC2InstanceManagementRole
   ```

   PowerShell

   ```
   Update-SSMServiceSetting `
   -SettingId "arn:aws:ssm:region:account-id:servicesetting/ssm/managed-instance/
   default-ec2-instance-management-role" `
   -SettingValue "service-role/AWSSystemsManagerDefaultEC2InstanceManagementRole"
   ```

   Il n'y a pas de sortie si la commande réussit.

5. Exécutez la commande suivante pour afficher les paramètres de service actuels pour la configuration de gestion d'hôte par défaut dans les versions actuelles Compte AWS et Région AWS.

   Linux & macOS

   ```
   aws ssm get-service-setting \
   --setting-id arn:aws:ssm:region:account-id:servicesetting/ssm/managed-instance/
   default-ec2-instance-management-role
   ```

   Windows

   ```
   aws ssm get-service-setting ^
   ```

```
--setting-id arn:aws:ssm:region:account-id:servicesetting/ssm/managed-instance/
default-ec2-instance-management-role
```

PowerShell

```
Get-SSMServiceSetting `
-SettingId "arn:aws:ssm:region:account-id:servicesetting/ssm/managed-instance/
default-ec2-instance-management-role"
```

La commande renvoie des informations telles que les suivantes.

```
{
    "ServiceSetting": {
        "SettingId": "/ssm/managed-instance/default-ec2-instance-management-role",
        "SettingValue": "service-role/
AWSSystemsManagerDefaultEC2InstanceManagementRole",
        "LastModifiedDate": "2022-11-28T08:21:03.576000-08:00",
        "LastModifiedUser": "System",
        "ARN": "arn:aws:ssm:us-east-2:-123456789012:servicesetting/ssm/managed-
instance/default-ec2-instance-management-role",
        "Status": "Custom"
    }
}
```

Désactivation du paramètre de configuration de gestion d'hôte par défaut

Vous pouvez désactiver la configuration de gestion d'hôte par défaut depuis la Fleet Manager console ou à l'aide du AWS Command Line Interface ou AWS Tools for Windows PowerShell.

Vous devez désactiver le paramètre de configuration de gestion d'hôte par défaut un par un dans chaque région où vous ne souhaitez plus que vos instances Amazon EC2 soient gérées par cette configuration. Le désactiver dans une région ne le désactive pas dans toutes les régions.

Si vous désactivez la configuration de gestion des hôtes par défaut et que vous n'avez pas associé de profil d'instance à vos instances Amazon EC2 autorisant l'accès à Systems Manager, celles-ci ne seront plus gérées par Systems Manager.

Pour désactiver la configuration de gestion des hôtes par défaut (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Choisissez Gestion de compte, Configuration de la gestion des hôtes par défaut.
4. Désactivez l'option Activer la Configuration de gestion des hôtes par défaut.
5. Choisissez Configurer pour désactiver la Configuration de gestion des hôtes par défaut.

Pour désactiver la configuration de gestion des hôtes par défaut (ligne de commande)

- Ouvrez le AWS CLI ou les outils pour Windows PowerShell et exécutez la commande suivante. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

  Linux & macOS

  ```
  aws ssm reset-service-setting \
  --setting-id arn:aws:ssm:region:account-id:servicesetting/ssm/managed-instance/
  default-ec2-instance-management-role
  ```

  Windows

  ```
  aws ssm reset-service-setting ^
  --setting-id arn:aws:ssm:region:account-id:servicesetting/ssm/managed-instance/
  default-ec2-instance-management-role
  ```

  PowerShell

  ```
  Reset-SSMServiceSetting `
  -SettingId "arn:aws:ssm:region:account-id:servicesetting/ssm/managed-instance/
  default-ec2-instance-management-role"
  ```

Exemples de politique du moindre privilège pour la configuration de gestion des hôtes par défaut

Les exemples de politiques suivants montrent comment empêcher les membres de votre organisation d'apporter des modifications au paramètre de configuration de gestion des hôtes par défaut dans votre compte.

Politique de contrôle des services pour AWS Organizations

La politique suivante explique comment empêcher les membres non administrateurs de mettre à jour votre paramètre AWS Organizations de configuration de gestion d'hôte par défaut. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

```
{
    "Version":"2012-10-17",
    "Statement":[
        {
            "Effect":"Deny",
            "Action":[
                "ssm:UpdateServiceSetting",
                "ssm:ResetServiceSetting"
            ],
            "Resource":"arn:aws:ssm:*:*:servicesetting/ssm/managed-instance/default-
ec2-instance-management-role",
            "Condition":{
                "StringNotEqualsIgnoreCase":{
                "aws:PrincipalTag/job-function":[
                    "administrator"
                ]
                }
            }
        },
        {
            "Effect":"Deny",
            "Action":[
                "iam:PassRole"
            ],
            "Resource":"arn:aws:iam::*:role/service-role/
AWSSystemsManagerDefaultEC2InstanceManagementRole",
            "Condition":{
                "StringEquals":{
                "iam:PassedToService":"ssm.amazonaws.com"
                },
```

```
                "StringNotEqualsIgnoreCase":{
                "aws:PrincipalTag/job-function":[
                    "administrator"
                ]
                }
            }
        },
        {
            "Effect":"Deny",
            "Resource":"arn:aws:iam::*:role/service-role/
AWSSystemsManagerDefaultEC2InstanceManagementRole",
            "Action":[
                "iam:AttachRolePolicy",
                "iam:DeleteRole"
            ],
            "Condition":{
                "StringNotEqualsIgnoreCase":{
                "aws:PrincipalTag/job-function":[
                    "administrator"
                ]
                }
            }
        }
    ]
}
```

Politique pour les principaux IAM

La politique suivante explique comment empêcher les groupes, les rôles ou les utilisateurs IAM de mettre à jour votre paramètre AWS Organizations de configuration de gestion d'hôte par défaut. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "ssm:UpdateServiceSetting",
                "ssm:ResetServiceSetting"
            ],
```

```
            "Resource": "arn:aws:ssm:region:account-id:servicesetting/ssm/managed-
instance/default-ec2-instance-management-role"
        },
        {
            "Effect": "Deny",
            "Action": [
                "iam:AttachRolePolicy",
                "iam:DeleteRole",
                "iam:PassRole"
            ],
            "Resource": "arn:aws:iam::account-id:role/service-role/
AWSSystemsManagerDefaultEC2InstanceManagementRole"
        }
    ]
}
```

Connexion à une instance Windows Server gérée à l'aide de Remote Desktop

Vous pouvez utiliser une fonctionnalité de Fleet Manager AWS Systems Manager, pour vous connecter à vos instances Windows Server Amazon Elastic Compute Cloud (Amazon EC2) à l'aide Remote Desktop Protocol du (RDP). Fleet Manager Le bureau à distance, qui est alimenté par NICE DCV, vous fournit une connectivité sécurisée à vos instances Windows Server directement depuis la console Systems Manager. Vous pouvez établir jusqu'à quatre connexions simultanées dans une seule fenêtre de navigateur.

Actuellement, vous ne pouvez utiliser le Bureau à distance qu'avec des instances exécutant Windows Server 2012 RTM ou une version ultérieure. Le Bureau à distance prend uniquement en charge la langue anglaise.

> **Note**
>
> Fleet ManagerRemote Desktop est un service réservé à la console et ne prend pas en charge les connexions en ligne de commande à vos instances gérées. Pour vous connecter à une instance Windows Server gérée via un shell, vous pouvez utiliser Session Manager une autre fonctionnalité de AWS Systems Manager. Pour plus d'informations, consultez AWS Systems Manager Session Manager.

Pour plus d'informations sur la configuration des autorisations AWS Identity and Access Management (IAM) permettant à vos instances d'interagir avec Systems Manager, consultez la section Configurer les autorisations d'instance pour Systems Manager.

Rubriques

Configuration de votre environnement

Avant d'utiliser le Bureau à distance, vérifiez que votre environnement respecte les conditions requises suivantes :

- Configuration des nœuds gérés

  Assurez-vous que vos instances Amazon EC2 sont configurées en tant que nœuds gérés dans Systems Manager.

- Version minimale de SSM Agent

  Vérifiez que les nœuds exécutent SSM Agent version 3.0.222.0 ou supérieure. Pour plus d'informations sur la vérification de la version de l'agent exécutée sur un nœud, veuillez consulter la rubrique Vérification du numéro de version de l'SSM Agent. Pour plus d'informations sur l'installation ou la mise à jour de SSM Agent, consultez Utilisation de l'option SSM Agent.

- Configuration du port RDP

  Pour accepter les connexions distantes, le service Remote Desktop Services sur vos nœuds Windows Server doit utiliser le port RDP 3389 par défaut. Il s'agit de la configuration par défaut sur Amazon Machine Images (AMIs) fournie par AWS. Vous n'êtes pas explicitement obligé d'ouvrir des ports entrants pour utiliser le Bureau à distance.

- Version du module PSReadLine pour les fonctionnalités du clavier

  Pour vous assurer que votre clavier fonctionne correctement dans PowerShell, vérifiez que la version 2.2.2 ou supérieure du module PSReadLine est installée sur les nœuds exécutant

Windows Server 2022. S'ils utilisent une version antérieure, vous pouvez installer la version requise à l'aide de la commande suivante.

```
Install-Module `
    -Name PSReadLine `
    -Repository PSGallery `-MinimumVersion 2.2.2
```

- Configuration de Session Manager

  Avant de pouvoir utiliser le Bureau à distance, vous devez remplir les conditions suivantes pour la configuration de Session Manager. Lorsque vous vous connectez à une instance à l'aide de Remote Desktop, toutes les préférences de session définies pour votre Compte AWS et Région AWS sont appliquées. Pour plus d'informations, consultez Configuration de Session Manager.

  > **Note**
  >
  > Si vous journalisez l'activité de Session Manager à l'aide d'Amazon Simple Storage Service (Amazon S3), vos connexions Bureau à distance génèrent l'erreur suivante dans `bucket_name/Port/stderr`. Cette erreur est prévue et peut être ignorée sans risque.
  >
  > ```
  > Setting up data channel with id SESSION_ID failed: failed to create websocket
  >  for datachannel with error: CreateDataChannel failed with no output or
  >  error: createDataChannel request failed: unexpected response from the service
  >  <BadRequest>
  > <ClientErrorMessage>Session is already terminated</ClientErrorMessage>
  > </BadRequest>
  > ```

Configuration des autorisations IAM pour le Bureau à distance

Outre les autorisations IAM requises pour Systems Manager et Session Manager, l'utilisateur ou le rôle que vous utilisez pour accéder à la console doit autoriser les actions suivantes :

- `ssm-guiconnect:CancelConnection`
- `ssm-guiconnect:GetConnection`
- `ssm-guiconnect:StartConnection`

Vous trouverez ci-dessous des exemples de politiques IAM que vous pouvez attacher à un utilisateur ou un rôle pour permettre différents types d'interaction avec le Bureau à distance. Remplacez chaque

example resource placeholder (espace réservé pour les ressources) avec vos propres informations.

Politique standard de connexion aux instances EC2

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "EC2",
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:GetPasswordData"
            ],
            "Resource": "*"
        },
        {
            "Sid": "SSM",
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeInstanceProperties",
                "ssm:GetCommandInvocation",
                "ssm:GetInventorySchema"
            ],
            "Resource": "*"
        },
        {
            "Sid": "TerminateSession",
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession"
            ],
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/aws:ssmmessages:session-id": [
                        "${aws:userid}"
                    ]
                }
            }
        },
        {
            "Sid": "SSMStartSession",
```

```
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ec2:*:account-id:instance/*",
                "arn:aws:ssm:*:account-id:managed-instance/*",
                "arn:aws:ssm:*::document/AWS-StartPortForwardingSession"
            ],
            "Condition": {
                "BoolIfExists": {
                    "ssm:SessionDocumentAccessCheck": "true"
                },
                "ForAnyValue:StringEquals": {
                    "aws:CalledVia": "ssm-guiconnect.amazonaws.com"
                }
            }
        },
        {
            "Sid": "GuiConnect",
            "Effect": "Allow",
            "Action": [
                "ssm-guiconnect:CancelConnection",
                "ssm-guiconnect:GetConnection",
                "ssm-guiconnect:StartConnection"
            ],
            "Resource": "*"
        }
    ]
}
```

Politique de connexion à des instances EC2 avec des balises spécifiques

> **Note**
>
> Dans la politique IAM suivante, la `SSMStartSession` section nécessite un Amazon Resource Name (ARN) pour l'`ssm:StartSession`action. Comme indiqué, l'ARN que vous spécifiez ne nécessite pas d' Compte AWS ID. Si vous spécifiez un identifiant de compte, Fleet Manager renvoie un`AccessDeniedException`.

La `AccessTaggedInstances` section, située plus bas dans l'exemple de politique, nécessite également des ARN pour`ssm:StartSession`. Pour ces ARN, vous spécifiez Compte AWS des identifiants.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "EC2",
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:GetPasswordData"
            ],
            "Resource": "*"
        },
        {
            "Sid": "SSM",
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeInstanceProperties",
                "ssm:GetCommandInvocation",
                "ssm:GetInventorySchema"
            ],
            "Resource": "*"
        },
        {
            "Sid": "SSMStartSession",
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ssm:*::document/AWS-StartPortForwardingSession"
            ],
            "Condition": {
                "BoolIfExists": {
                    "ssm:SessionDocumentAccessCheck": "true"
                },
                "ForAnyValue:StringEquals": {
                    "aws:CalledVia": "ssm-guiconnect.amazonaws.com"
```

```
                }
            }
        },
        {
            "Sid": "AccessTaggedInstances",
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ec2:*:account-id:instance/*",
                "arn:aws:ssm:*:account-id:managed-instance/*"
            ],
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/tag key": [
                        "tag value"
                    ]
                }
            }
        },
        {
            "Sid": "GuiConnect",
            "Effect": "Allow",
            "Action": [
                "ssm-guiconnect:CancelConnection",
                "ssm-guiconnect:GetConnection",
                "ssm-guiconnect:StartConnection"
            ],
            "Resource": "*"
        }
    ]
}
```

Politique permettant AWS IAM Identity Center aux utilisateurs de se connecter aux instances EC2

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

```
                "sso:ListDirectoryAssociations*",
                "identitystore:DescribeUser"
            ],
            "Resource": "*"
        },
        {
            "Sid": "EC2",
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:GetPasswordData"
            ],
            "Resource": "*"
        },
        {
            "Sid": "SSM",
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeInstanceProperties",
                "ssm:GetCommandInvocation",
                "ssm:GetInventorySchema"
            ],
            "Resource": "*"
        },
        {
            "Sid": "TerminateSession",
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession"
            ],
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/aws:ssmmessages:session-id": [
                        "${aws:userName}"
                    ]
                }
            }
        },
        {
            "Sid": "SSMStartSession",
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
```

```
            ],
            "Resource": [
                "arn:aws:ec2:*:*:instance/*",
                "arn:aws:ssm:*:*:managed-instance/*",
                "arn:aws:ssm:*:*:document/AWS-StartPortForwardingSession"
            ],
            "Condition": {
                "BoolIfExists": {
                    "ssm:SessionDocumentAccessCheck": "true"
                },
                "ForAnyValue:StringEquals": {
                    "aws:CalledVia": "ssm-guiconnect.amazonaws.com"
                }
            }
        },
        {
            "Sid": "SSMSendCommand",
            "Effect": "Allow",
            "Action": [
                "ssm:SendCommand"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:instance/*",
                "arn:aws:ssm:*:*:managed-instance/*",
                "arn:aws:ssm:*:*:document/AWSSSO-CreateSSOUser"
            ],
            "Condition": {
                "BoolIfExists": {
                    "ssm:SessionDocumentAccessCheck": "true"
                }
            }
        },
        {
            "Sid": "GuiConnect",
            "Effect": "Allow",
            "Action": [
                "ssm-guiconnect:CancelConnection",
                "ssm-guiconnect:GetConnection",
                "ssm-guiconnect:StartConnection"
            ],
            "Resource": "*"
        }
    ]
```

```
}
```

Authentification des connexions Bureau à distance

Lorsque vous établissez une connexion distante, vous pouvez vous authentifier à l'aide des informations d'identification Windows ou de la paire de clés Amazon EC2 (fichier `.pem`) associée à l'instance. Pour plus d'informations sur l'utilisation des paires de clés, consultez les paires de clés et Windows instances Amazon EC2 dans le guide de l'utilisateur Amazon EC2.

Sinon, si vous êtes authentifié auprès de l'utilisateur AWS IAM Identity Center, AWS Management Console vous pouvez vous connecter à vos instances sans fournir d'informations d'identification supplémentaires. Pour obtenir un exemple de politique permettant l'authentification des connexions distantes à l'aide d'IAM Identity Center, veuillez consulter la rubrique Configuration des autorisations IAM pour le Bureau à distance.

Avant de commencer

Veuillez tenir compte des conditions suivantes pour l'utilisation de l'authentification IAM Identity Center avant de commencer à vous connecter via le Bureau à distance.

- Le Bureau à distance prend en charge l'authentification IAM Identity Center pour les nœuds dans la Région AWS dans laquelle vous avez activé IAM Identity Center.
- Le Bureau à distance prend en charge les noms d'utilisateur IAM Identity Center comportant jusqu'à 16 caractères.
- Le Bureau à distance prend en charge les noms d'utilisateur IAM Identity Center comportant des caractères alphanumériques et les caractères spéciaux suivants : `. - _`

  > **Important**
  >
  > Les connexions échoueront pour les noms d'utilisateur IAM Identity Center qui contiennent les caractères suivants : `+ = , @`.
  > IAM Identity Center prend en charge ces caractères dans les noms d'utilisateur, mais pas les connexions RDP Fleet Manager.

- Lorsqu'une connexion est authentifiée à l'aide d'IAM Identity Center, le Bureau à distance crée un utilisateur local Windows dans le groupe d'administrateurs locaux de l'instance. Cet utilisateur persiste après la fin de la connexion distante.
- Le Bureau à distance n'autorise pas l'authentification IAM Identity Center pour les nœuds qui sont des contrôleurs de domaine Microsoft Active Directory.

- Bien que le Bureau à distance vous permette d'utiliser l'authentification IAM Identity Center pour les nœuds joints à un domaine Active Directory, nous vous déconseillons de le faire. Cette méthode d'authentification accorde aux utilisateurs des autorisations administratives qui peuvent remplacer les autorisations plus restrictives accordées par le domaine.

Régions prises en charge pour l'authentification IAM Identity Center

Les connexions Remote Desktop qui utilisent l'authentification IAM Identity Center sont prises en charge dans les Régions AWS suivantes :

- USA Est (Ohio) (us-east-2)
- USA Est (Virginie du Nord) (us-east-1)
- US Ouest (N. California) (us-west-1)
- USA Ouest (Oregon) (us-west-2)
- Afrique (Le Cap) (af-south-1)
- Asie-Pacifique (Hong Kong) (ap-east-1)
- Asie-Pacifique (Mumbai) (ap-south-1)
- Asie-Pacifique (Tokyo) (ap-northeast-1)
- Asie-Pacifique (Séoul) (ap-northeast-2)
- Asie-Pacifique (Osaka) (ap-northeast-3)
- Asie-Pacifique (Singapour) (ap-southeast-1)
- Asie-Pacifique (Sydney) (ap-southeast-2)
- Asie-Pacifique (Jakarta) (ap-southeast-3)
- Canada (Centre) (ca-central-1)
- Europe (Francfort) (eu-central-1)
- Europe (Stockholm) (eu-north-1)
- Europe (Irlande) (eu-west-1)
- Europe (Londres) (eu-west-2)
- Europe (Paris) (eu-west-3)
- Israël (Tel Aviv) (il-central-1)
- Amérique du Sud (São Paulo) (sa-east-1)
- Europe (Milan) (eu-south-1)

- Moyen-Orient (Bahreïn) (me-south-1)
- AWS GovCloud (US-Est) (us-gov-east-1)
- AWS GovCloud (US-Ouest) (us-gov-west-1)

Durée et simultanéité des connexions distantes

Les conditions suivantes s'appliquent aux connexions Bureau à distance actives :

- Durée de connexion

  Par défaut, une connexion Bureau à distance est déconnectée au bout de 60 minutes. Pour empêcher la déconnexion d'une connexion, vous pouvez choisir Renouveler la session avant d'être déconnecté pour réinitialiser la durée.
- Délai de connexion

  Une connexion Bureau à distance se déconnecte après plus de 10 minutes d'inactivité.
- Connexions simultanées

  Par défaut, vous pouvez avoir un maximum de 5 connexions Remote Desktop actives à la fois pour le même Compte AWS et Région AWS. Pour demander une augmentation du quota de service allant jusqu'à 25 connexions simultanées, veuillez consulter la rubrique Demande d'augmentation de quota dans le Guide de l'utilisateur Service Quotas.

Connexion à un nœud géré à l'aide du Bureau à distance

Support du copier-coller du texte dans le navigateur

À l'aide des navigateurs Google Chrome et Microsoft Edge, vous pouvez copier et coller du texte d'un nœud géré vers votre machine locale, et de votre machine locale vers un nœud géré auquel vous êtes connecté.

À l'aide du navigateur Mozilla Firefox, vous pouvez copier et coller du texte depuis un nœud géré vers votre ordinateur local uniquement. La copie depuis votre machine locale vers le nœud géré n'est pas prise en charge.

Pour vous connecter à un nœud géré à l'aide de Fleet Manager Bureau à distance

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.

2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Choisissez le nœud auquel vous souhaitez vous connecter. Vous pouvez sélectionner la case à cocher ou le nom du nœud.
4. Dans le menu Actions du nœud, sélectionnez Se connecter au Bureau à distance.
5. Sélectionnez votre type d'authentification préféré dans le champ Authentication type (Type d'authentification). Si vous choisissez Informations d'identification utilisateur, saisissez le nom d'utilisateur et le mot de passe d'un compte utilisateur Windows sur le nœud auquel vous vous connectez. Si vous choisissez Paire de clés, vous pouvez fournir l'authentification à l'aide d'une des méthodes suivantes :

   a. Choisissez Parcourir la machine locale si vous souhaitez sélectionner la clé PEM associée à votre instance dans votre système de fichiers local.

      - ou -

   b. Choisissez Coller le contenu de la paire de clés si vous souhaitez copier le contenu du fichier PEM et le coller dans le champ prévu à cet effet.
6. Cliquez sur Connect (Connexion).
7. Pour choisir votre résolution d'affichage préférée, dans le menu Actions, choisissez Resolutions (Résolutions), puis sélectionnez l'une des options suivantes :

   - Adaptation automatique
   - 1920 x 1080
   - 1400 x 900
   - 1#366 x 768
   - 800 x 600

   L'option Adapt Automatically (Adapter automatiquement) définit la résolution en fonction de la taille d'écran détectée.

Gestion des volumes Amazon EBS sur des instances gérées

Amazon Elastic Block Store (Amazon EBS) fournit des volumes de stockage par bloc à utiliser avec les instances Amazon Elastic Compute Cloud (EC2). Les volumes EBS se comportent comme des périphériques de stockage en mode bloc bruts non formatés. Vous pouvez monter ces volumes en tant qu'appareils sur vos instances.

Vous pouvez utiliser Fleet Manager une fonctionnalité de AWS Systems Manager pour gérer les volumes Amazon EBS sur vos instances gérées. Par exemple, vous pouvez initialiser un volume EBS, formater une partition et monter le volume pour le rendre utilisable.

> **Note**
>
> Fleet Manager ne prend actuellement en charge la gestion des volumes Amazon EBS que pour les instances Windows Server.

Affichage des détails d'un volume

Pour afficher les détails d'un volume EBS avec Fleet Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Choisissez le bouton situé en regard de l'instance gérée dont vous voulez afficher les détails de volume EBS.
4. Sélectionnez Afficher les détails.
5. Choisissez Outils, Volumes EBS.
6. Pour afficher les détails d'un volume EBS, choisissez son ID dans la colonne ID du volume.

Initialisation et formatage d'un volume EBS

Pour initialiser et formater un volume EBS avec Fleet Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Choisissez le bouton situé en regard de l'instance gérée pour laquelle vous souhaitez initialiser, formater et monter un volume EBS. Vous ne pouvez initialiser un volume EBS que si son disque est vide.
4. Sélectionnez Afficher les détails.
5. Dans le menu Outils, sélectionnez Volumes EBS.
6. Choisissez le bouton en regard du volume EBS que vous voulez initialiser et formater.

7. Choisissez Initialiser et formater.
8. Dans Style de partition, choisissez le style de partition que vous souhaitez utiliser pour le volume EBS.
9. (Facultatif) Choisissez une lettre de lecteur pour la partition.
10. (Facultatif) Saisissez un Nom de partition pour identifier la partition.
11. Choisissez le Système de fichiers à utiliser pour organiser les fichiers et les données stockés dans la partition.
12. Cliquez sur Confirmer pour rendre le volume EBS disponible à l’utilisation. Vous ne pouvez pas modifier la configuration de la partition depuis l’ AWS Management Console après confirmation, mais vous pouvez utiliser SSH ou RDP pour vous connecter à l’instance afin de modifier la configuration de la partition.

Utilisation du système de fichiers

Vous pouvez utiliser Fleet Manager une fonctionnalité de AWS Systems Manager pour travailler avec le système de fichiers sur vos nœuds gérés. À l'aide de Fleet Manager, vous pouvez consulter des informations sur les données de répertoire et de fichier stockées sur les volumes attachés à vos nœuds gérés. Par exemple, vous pouvez afficher le nom, la taille, l'extension, le propriétaire et les autorisations de vos répertoires et fichiers. Vous pouvez prévisualiser jusqu'à 10 000 lignes de données de fichier sous forme de texte depuis la console Fleet Manager. Vous pouvez également utiliser cette fonction pour des fichiers `tail`. Lorsque vous utilisez `tail` pour afficher des données de fichier, les 10 dernières lignes du fichier s'affichent initialement. À mesure que de nouvelles lignes de données sont écrites dans le fichier, l'affichage se met à jour en temps réel. Vous pouvez donc consulter les données de journal depuis la console, ce qui peut accroître votre efficacité de résolution des problèmes et d'administration des systèmes. En outre, vous pouvez créer des répertoires et copier, couper, coller, renommer ou supprimer des fichiers et des répertoires.

Nous vous recommandons de créer des sauvegardes régulières ou de prendre des instantanés des volumes Amazon Elastic Block Store (Amazon EBS) attachés à vos nœuds gérés. Lorsque vous copiez ou coupez et collez des fichiers, les fichiers et répertoires existants de l'emplacement de destination qui portent le même nom que les nouveaux fichiers ou répertoires sont remplacés. De graves problèmes peuvent survenir si vous remplacez ou modifiez des fichiers et des répertoires système. AWS ne garantit pas que ces problèmes puissent être résolus. Toute modification apportée aux fichiers système se fait à vos propres risques. Vous êtes responsable de toutes les modifications que vous apportez aux fichiers et répertoires, et vous devez vous assurer que vous disposez de

sauvegardes de ceux-ci. La suppression ou le remplacement de fichiers et de répertoires ne peut pas être annulée.

> **Note**
>
> Fleet ManagerutiliseSession Manager, une fonctionnalité de AWS Systems Manager, pour afficher des aperçus de texte et des `tail` fichiers. Pour les instances Amazon Elastic Compute Cloud (Amazon EC2), le profil d'instance attaché à vos instances gérées doit fournir à Session Manager les autorisations nécessaires à l'utilisation de cette fonction. Pour de plus amples informations sur l'ajout d'autorisations Session Manager à un profil d'instance, veuillez consulter Ajout d'autorisations Session Manager à un rôle IAM existant. Vous devez également activer le chiffrement AWS Key Management Service (AWS KMS) dans vos préférences de session pour utiliser des fonctions Fleet Manager. Pour plus d'informations sur l'activation du AWS KMS chiffrement pourSession Manager, consultezActiver le chiffrement des données de session par clé KMS (console).

Pour afficher le système de fichiers avec Fleet Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Sélectionnez le lien du nœud géré associé au système de fichiers que vous souhaitez afficher.
4. Choisissez Outils, Système de fichiers.

Pour afficher des aperçus de texte de fichiers avec Fleet Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Sélectionnez le lien du nœud géré associé aux fichiers que vous souhaitez prévisualiser.
4. Choisissez Outils, Système de fichiers.
5. Dans le champ File name (Nom de fichier), sélectionnez le nom du répertoire qui contient le fichier que vous souhaitez prévisualiser.
6. Cliquez sur le bouton en regard du fichier dont vous voulez prévisualiser le contenu.
7. Choisissez Actions, Aperçu sous forme de texte.

Pour mettre à la file des fichiers avec Fleet Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Sélectionnez le lien du nœud géré associé aux fichiers que vous souhaitez mettre en file.
4. Choisissez Outils, Système de fichiers.
5. Dans le champ File name (Nom de fichier), sélectionnez le nom du répertoire qui contient le fichier que vous souhaitez mettre en file.
6. Cliquez sur le bouton en regard du fichier dont vous voulez mettre en file le contenu.
7. Choisissez Actions, Fichier en queue.

Pour copier ou couper et coller des fichiers ou des répertoires avec Fleet Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Sélectionnez le lien du nœud géré associé aux fichiers que vous souhaitez copier ou couper et coller.
4. Choisissez Outils, Système de fichiers.
5. Pour copier ou couper un fichier, dans le champ File name (Nom de fichier), choisissez le nom du répertoire qui contient le fichier à copier ou couper. Pour copier ou couper un répertoire, choisissez le bouton situé en regard du répertoire que vous souhaitez copier ou couper, puis passez à l'étape 8.
6. Cliquez sur le bouton situé en regard du fichier que vous souhaitez copier ou couper.
7. Dans le menu Actions, sélectionnez Copy (Copier) ou Cut (Couper).
8. Dans File system (Système de fichiers), choisissez le bouton situé en regard du répertoire dans lequel vous souhaitez coller le fichier.
9. Dans le menu Actions, sélectionnez Paste (Coller).

Pour renommer des fichiers ou des répertoires avec Fleet Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.

2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Sélectionnez le lien du nœud géré associé aux fichiers ou répertoires que vous souhaitez renommer.
4. Choisissez Outils, Système de fichiers.
5. Pour renommer un fichier, dans le champ File name (Nom de fichier), sélectionnez le nom du répertoire qui contient le fichier que vous souhaitez renommer. Pour renommer un répertoire, cliquez sur le bouton situé en regard du répertoire que vous souhaitez renommer, puis passez à l'étape 8.
6. Cliquez sur le bouton situé en regard du fichier dont vous souhaitez renommer le contenu.
7. Choisissez Actions, Renommer.
8. Dans le champ Nom du fichier, saisissez le nouveau nom de fichier et sélectionnez Renommer.

Pour supprimer des fichiers ou des répertoires avec Fleet Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Sélectionnez le lien du nœud géré associé aux fichiers ou répertoires que vous souhaitez supprimer.
4. Choisissez Outils, Système de fichiers.
5. Pour supprimer un fichier, dans le champ File name (Nom de fichier), sélectionnez le nom du répertoire qui contient le fichier que vous souhaitez supprimer. Pour supprimer un répertoire, cliquez sur le bouton situé en regard du répertoire que vous souhaitez supprimer, puis passez à l'étape 7.
6. Cliquez sur le bouton situé en regard du fichier dont vous souhaitez supprimer le contenu.
7. Sélectionnez Actions, Supprimer.

Pour créer un annuaire avec Fleet Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Sélectionnez le lien du nœud géré dans lequel vous souhaitez créer un répertoire.

4. Choisissez Outils, Système de fichiers.
5. Sélectionnez le File name (Nom de fichier) du répertoire où vous souhaitez créer un nouveau répertoire.
6. Sélectionnez Create directory (Créer un répertoire).
7. Dans le champ Nom du répertoire, saisissez le nom du nouveau répertoire et sélectionnez Créer le répertoire.

Surveillance de la performance des nœuds gérés

Vous pouvez utiliser Fleet Manager une fonctionnalité de pour afficher AWS Systems Manager les données de performance relatives à vos nœuds gérés en temps réel. Les données de performance sont extraites des compteurs de performance.

Les compteurs de performance suivants sont disponibles dans Fleet Manager :

- Utilisation de l'UC
- Utilisation des entrées/sorties (I/O) de disque
- Trafic réseau
- Utilisation de la mémoire

> **Note**
>
> Fleet ManagerutiliseSession Manager, une capacité de AWS Systems Manager, pour récupérer des données de performance. Pour les instances Amazon Elastic Compute Cloud (Amazon EC2), le profil d'instance attaché à vos instances gérées doit fournir à Session Manager les autorisations nécessaires à l'utilisation de cette fonction. Pour de plus amples informations sur l'ajout d'autorisations Session Manager à un profil d'instance, veuillez consulter Ajout d'autorisations Session Manager à un rôle IAM existant. Vous devez également activer le chiffrement AWS Key Management Service (AWS KMS) dans vos préférences de session pour utiliser des fonctions Fleet Manager. Pour plus d'informations sur l'activation du AWS KMS chiffrement pourSession Manager, consultezActiver le chiffrement des données de session par clé KMS (console).

Pour consulter les données de performances avec Fleet Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Cliquez sur le bouton situé en regard du nœud géré dont vous souhaitez surveiller la performance.
4. Sélectionnez Afficher les détails.
5. Choisissez Outils, Compteurs de performance.

Utilisation des processus

Vous pouvez utiliser Fleet Manager une fonctionnalité de AWS Systems Manager pour travailler avec des processus sur vos instances gérées. À l'aide de Fleet Manager, vous pouvez consulter des informations sur les processus. Par exemple, vous pouvez visualiser l'utilisation du processeur et de la mémoire des processus en plus de leurs handles et threads. Avec Fleet Manager, vous pouvez démarrer et résilier des processus à partir de la console.

> **Note**
>
> Fleet ManagerutiliseSession Manager, une capacité de AWS Systems Manager, pour récupérer des données de processus. Pour les instances Amazon Elastic Compute Cloud (Amazon EC2), le profil d'instance attaché à vos instances gérées doit fournir à Session Manager les autorisations nécessaires à l'utilisation de cette fonction. Pour de plus amples informations sur l'ajout d'autorisations Session Manager à un profil d'instance, veuillez consulter Ajout d'autorisations Session Manager à un rôle IAM existant. Vous devez également activer le chiffrement AWS Key Management Service (AWS KMS) dans vos préférences de session pour utiliser des fonctions Fleet Manager. Pour plus d'informations sur l'activation du AWS KMS chiffrement pourSession Manager, consultezActiver le chiffrement des données de session par clé KMS (console).

Pour afficher les détails relatifs aux processus à l'aide de Fleet Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.

3. Sélectionnez le lien de l'instance dont vous souhaitez visualiser les processus.
4. Choisissez Outils, Processus.

Pour démarrer un processus avec Fleet Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Sélectionnez le lien de l'instance sur laquelle vous souhaitez démarrer un processus.
4. Choisissez Outils, Processus.
5. Sélectionnez Start new process (Démarrer un nouveau processus).
6. Dans le champ Nom du processus ou chemin d'accès complet, saisissez le nom du processus ou le chemin d'accès complet à l'exécutable.
7. (Facultatif) Dans le champ Répertoire de travail, saisissez le chemin d'accès au répertoire dans lequel vous souhaitez que le processus s'exécute.

Pour arrêter un processus avec Fleet Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Sélectionnez le lien de l'instance sur laquelle vous souhaitez démarrer un processus.
4. Choisissez Outils, Processus.
5. Cliquez sur le bouton situé en regard du processus que vous souhaitez arrêter.
6. Sélectionnez Actions, Terminer le processus ou Actions, Arrêter l'arborescence du processus.

> **Note**
>
> L'arrêt d'une arborescence de processus met également fin à tous les processus et applications qui utilisent ce processus.

Afficher les journaux sur les nœuds gérés

Vous pouvez utiliser Fleet Manager une fonctionnalité de pour afficher AWS Systems Manager les données de journal stockées sur vos nœuds gérés. Concernant les nœuds gérés Windows, vous pouvez afficher les journaux d'événements Windows et copier leurs détails depuis la console. Pour faciliter la recherche d'événements, filtrez les journaux d'événements Windows par Niveau d'événement, ID d'événement, Source de l'événement et Heure de création. Vous pouvez également afficher d'autres données du journal via la procédure d'affichage du système de fichiers. Pour de plus amples informations sur l'affichage du système de fichiers avec Fleet Manager, veuillez consulter Utilisation du système de fichiers.

Pour afficher les journaux d'événements Windows avec Fleet Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Cliquez sur le bouton situé en regard du nœud géré dont vous souhaitez visualiser les journaux d'événements.
4. Sélectionnez Afficher les détails.
5. Choisissez Outils, Journaux d'événements Windows.
6. Sélectionnez le Nom de journal contenant les événements que vous voulez afficher.
7. Cliquez sur le bouton en regard du Nom de journal que vous voulez afficher, puis sélectionnez Afficher les événements.
8. Cliquez sur le bouton en regard de l'événement que vous voulez afficher, puis sélectionnez Afficher les détails de l'événement.
9. (Facultatif) Sélectionnez Copier en tant que JSON pour copier les détails de l'événement dans le presse-papier.

Gestion des comptes utilisateur du système d'exploitation sur les nœuds gérés

Vous pouvez utiliser Fleet Manager une fonctionnalité de AWS Systems Manager pour gérer les comptes utilisateur du système d'exploitation (OS) sur vos nœuds gérés. Par exemple, vous pouvez créer et supprimer des utilisateurs et des groupes. En outre, vous pouvez afficher des détails tels que l'appartenance à un groupe, des rôles utilisateur et un statut.

> **Important**
>
> Fleet Managerutilisations Run Command et Session Manager fonctionnalités de AWS Systems Manager, pour diverses opérations de gestion des utilisateurs. Par conséquent, un utilisateur peut octroyer des autorisations à un compte utilisateur du système d'exploitation, ce qui ne serait pas possible autrement. Cela est dû au fait que AWS Systems Manager Agent (SSM Agent) s'exécute sur des instances Amazon Elastic Compute Cloud (Amazon EC2) à l'aide des autorisations root (Linux) ou des autorisations SYSTEM (Windows Server). Pour de plus amples informations sur les restrictions d'accès aux commandes de niveau racine via l'SSM Agent, veuillez consulter Limitation de l'accès aux commandes de niveau racine via l'SSM Agent. Pour restreindre l'accès à cette fonctionnalité, nous vous recommandons de créer des politiques AWS Identity and Access Management (IAM) pour vos utilisateurs qui autorisent uniquement l'accès aux actions que vous définissez. Pour plus d'informations sur la création des politiques IAM pour Fleet Manager, consultez Étape 1 : créer une politique IAM avec des autorisations Fleet Manager.

Créer un groupe d'utilisateurs ou un utilisateur

> **Note**
>
> Fleet Manager utilise Session Manager pour définir des mots de passe pour les nouveaux utilisateurs. Concernant les instances Amazon EC2, le profil d'instance attaché à vos nœuds gérés doit fournir à Session Manager les autorisations nécessaires pour utiliser cette fonction. Pour de plus amples informations sur l'ajout d'autorisations Session Manager à un profil d'instance, veuillez consulter Ajout d'autorisations Session Manager à un rôle IAM existant. En outre, le chiffrement AWS Key Management Service (AWS KMS) doit être activé dans vos préférences de session pour utiliser les Fleet Manager fonctionnalités. Pour plus d'informations sur l'activation du chiffrement AWS KMS pour Session Manager, consultez Activer le chiffrement des données de session par clé KMS (console).

Pour créer un compte utilisateur SE avec Fleet Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.

3. Cliquez sur le bouton situé en regard du nœud géré sur lequel vous souhaitez créer un nouvel utilisateur.
4. Sélectionnez Afficher les détails.
5. Choisissez Outils, Utilisateurs et groupes.
6. Sélectionnez l'onglet Users (Utilisateurs), puis Create user (Créer un utilisateur).
7. Saisissez une valeur pour le Nom du nouvel utilisateur.
8. (Recommandé) Cochez la case en regard de Définir mot de passe. Au terme de la procédure, vous serez invité à fournir un mot de passe pour le nouvel utilisateur.
9. Sélectionnez Create user (Créer un utilisateur). Si vous avez coché la case de création d'un mot de passe pour le nouvel utilisateur, vous serez invité à saisir une valeur pour le mot de passe et à sélectionner Terminé. Si le mot de passe que vous spécifiez ne répond pas aux exigences spécifiées par les politiques locales ou par les politiques de domaine de votre nœud géré, une erreur est renvoyée.

Pour créer un groupe OS avec Fleet Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Cliquez sur le bouton situé en regard du nœud géré dans lequel vous souhaitez créer un groupe.
4. Sélectionnez Afficher les détails.
5. Choisissez Outils, Utilisateurs et groupes.
6. Sélectionnez l'onglet Groups (Groupes), puis Create group (Créer un groupe).
7. Saisissez une valeur pour le Nom du nouveau groupe.
8. (Facultatif) Saisissez une valeur pour la Description du nouveau groupe.
9. (Facultatif) Sélectionnez les utilisateurs à ajouter aux Membres du groupe du nouveau groupe.
10. Sélectionnez Create group (Créer un groupe).

Mise à jour de l'appartenance d'un utilisateur ou d'un groupe

Pour ajouter un compte utilisateur OS à un nouveau groupe avec Fleet Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.

2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Cliquez sur le bouton situé en regard du nœud géré contenant le compte utilisateur que vous souhaitez mettre à jour.
4. Sélectionnez Afficher les détails.
5. Choisissez Outils, Utilisateurs et groupes.
6. Sélectionnez l'onglet Utilisateurs.
7. Cliquez sur le bouton en regard de l'utilisateur que vous voulez mettre à jour.
8. Choisissez Actions, Ajouter un utilisateur au groupe.
9. Sélectionnez le groupe auquel vous voulez ajouter l'utilisateur sous Ajouter à un groupe.
10. Sélectionnez Ajouter un utilisateur à un groupe.

Pour modifier l'appartenance d'un groupe OS avec Fleet Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Cliquez sur le bouton situé en regard du nœud géré contenant le groupe que vous souhaitez mettre à jour.
4. Sélectionnez Afficher les détails.
5. Choisissez Outils, Utilisateurs et groupes.
6. Cliquez sur l'onglet Groups (Groupes).
7. Cliquez sur le bouton en regard du groupe que vous voulez mettre à jour.
8. Choisissez Actions, Modifier le groupe.
9. Sélectionnez les utilisateurs que vous voulez ajouter ou supprimer sous Membres du groupe.
10. Sélectionnez Modifier un groupe.

Supprimer un utilisateur ou un groupe d'utilisateurs

Pour supprimer un compte utilisateur OS avec Fleet Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.

3. Cliquez sur le bouton situé en regard du nœud géré contenant le compte utilisateur que vous souhaitez supprimer.
4. Sélectionnez Afficher les détails.
5. Choisissez Outils, Utilisateurs et groupes.
6. Sélectionnez l'onglet Utilisateurs.
7. Cliquez sur le bouton en regard de l'utilisateur que vous voulez supprimer.
8. Choisissez Actions, Supprimer l'utilisateur local.

Pour supprimer un groupe OS avec Fleet Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Cliquez sur le bouton situé en regard du nœud géré contenant le groupe que vous souhaitez supprimer.
4. Sélectionnez Afficher les détails.
5. Choisissez Outils, Utilisateurs et groupes.
6. Cliquez sur l'onglet Group (Groupe).
7. Cliquez sur le bouton en regard du groupe que vous voulez mettre à jour.
8. Choisissez Actions, Supprimer le groupe local.

Gestion du registre Windows sur les nœuds gérés

Vous pouvez utiliser Fleet Manager une fonctionnalité de AWS Systems Manager pour gérer le registre sur vos nœuds Windows Server gérés. Depuis la Fleet Manager, vous pouvez créer, copier, mettre à jour et supprimer des entrées et des valeurs de registre.

> **Important**
>
> Avant de modifier le registre, nous vous recommandons de créer une sauvegarde de celui-ci ou de prendre un instantané du volume Amazon Elastic Block Store (Amazon EBS) racine attaché à votre nœud géré. Une modification incorrecte du registre peut entraîner de graves problèmes. Ces problèmes peuvent vous obliger à réinstaller le système d'exploitation ou à restaurer le volume racine de votre nœud à partir d'un instantané. AWS ne garantit pas

que ces problèmes puissent être résolus. Vous modifiez le registre à vos propres risques. Vous êtes responsable de toutes les modifications apportées au registre et vous devez vous assurer de disposer de sauvegardes.

Créer une clé ou une entrée de registre Windows

Pour créer une clé de registre Windows avec Fleet Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Cliquez sur le bouton situé en regard du nœud géré sur lequel vous souhaitez créer une clé de registre.
4. Sélectionnez Afficher les détails.
5. Choisissez Outils, Registre Windows.
6. Sélectionnez le programme Hive dans lequel vous voulez créer une nouvelle clé de registre en sélectionnant le Nom de registre.
7. Choisissez Créer, Créer une clé de registre.
8. Cliquez sur le bouton en regard de l'entrée de registre dans laquelle vous voulez créer une nouvelle clé.
9. Sélectionnez Créer une clé de registre.
10. Saisissez une valeur pour le paramètre Nom de la nouvelle clé de Registre, puis sélectionnez Envoyer.

Pour créer une entrée de registre Windows avec Fleet Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Cliquez sur le bouton en regard de l'instance sur laquelle vous voulez créer une entrée de registre.
4. Sélectionnez Afficher les détails.
5. Choisissez Outils, Registre Windows.

6. Sélectionnez le programme Hive et la clé de registre suivante dans laquelle vous voulez créer une nouvelle entrée de registre en sélectionnant le Nom de registre.
7. Choisissez Créer, Créer une entrée de registre.
8. Saisissez une valeur pour le Name (Nom) de la nouvelle entrée de registre.
9. Sélectionnez le Type de valeur que vous voulez créer pour l'entrée de registre. Pour de plus amples informations sur les types de valeur de registre, veuillez consulter Types de valeurs de registre.
10. Saisissez une valeur pour la Valeur de la nouvelle entrée de registre.

Mettre à jour une entrée de Registre Windows

Pour mettre à jour une entrée de registre Windows avec Fleet Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Cliquez sur le bouton situé en regard du nœud géré sur lequel vous souhaitez mettre à jour une entrée de registre.
4. Sélectionnez Afficher les détails.
5. Choisissez Outils, Registre Windows.
6. Sélectionnez le programme Hive et la clé de registre suivante que vous voulez mettre à jour en sélectionnant le Nom de registre.
7. Cliquez sur le bouton en regard de l'entrée de registre que vous voulez mettre à jour.
8. Choisissez Actions, Mettre à jour l'entrée de registre.
9. Saisissez la nouvelle valeur pour la Value (Valeur) de l'entrée de registre.
10. Choisissez Mettre à jour.

Supprimer une entrée ou une clé de registre Windows

Pour supprimer une clé de registre Windows avec Fleet Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.

3. Cliquez sur le bouton situé en regard du nœud géré sur lequel vous souhaitez supprimer une clé de registre.
4. Choisissez Outils, Registre Windows.
5. Sélectionnez le programme Hive et la clé de registre suivante que vous voulez supprimer en sélectionnant le Nom de registre.
6. Cliquez sur le bouton en regard de la clé de registre que vous voulez supprimer.
7. Choisissez Actions, Supprimer la clé de registre.

Pour supprimer une entrée de registre Windows avec Fleet Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Cliquez sur le bouton situé en regard du nœud géré sur lequel vous souhaitez supprimer une entrée de registre.
4. Sélectionnez Afficher les détails.
5. Choisissez Outils, Registre Windows.
6. Sélectionnez le programme Hive et la clé de registre suivante contenant l'entrée que vous voulez supprimer en sélectionnant le Nom du registre.
7. Cliquez sur le bouton en regard de l'entrée de registre que vous voulez supprimer.
8. Choisissez Actions, Supprimer l'entrée du registre.

Accès au portail de la base de connaissances Red Hat

Vous pouvez utiliser Fleet Manager une fonctionnalité pour accéder au portail de AWS Systems Manager la base de connaissances si vous êtes un client Red Hat. Vous êtes considéré comme un client Red Hat si vous exécutez des instances Red Hat Enterprise Linux (RHEL) ou utilisez des services RHEL sur AWS. Le portail de la base de connaissances comprend des binaires, ainsi que des forums de partage de connaissances et de discussion destinés à assister la communauté, qui ne sont accessibles qu'aux clients sous licence Red Hat.

Outre les autorisations AWS Identity and Access Management (IAM) requises pour Systems ManagerFleet Manager, l'utilisateur ou le rôle que vous utilisez pour accéder à la console doit autoriser l'`rhelkb:GetRhelURL`action à accéder au portail de la base de connaissances.

Pour accéder au portail de la base de connaissances Red Hat

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Sélectionnez l'instance RHEL que vous souhaitez utiliser pour vous connecter au portail de la base de connaissances Red Hat.
4. Choisissez Gestion de compte, Accéder à la base de connaissances Red Hat pour ouvrir la page de la base de connaissances Red Hat.

Si vous utilisez RHEL on AWS pour exécuter des RHEL charges de travail entièrement prises en charge, vous pouvez également accéder à la base de connaissances Red Hat via le site Web de Red Hat en utilisant vos AWS informations d'identification.

Résolution des problèmes de disponibilité des nœuds gérés

Pour plusieurs AWS Systems Manager fonctionnalités telles que Run CommandDistributor, etSession Manager, vous pouvez choisir de sélectionner manuellement les nœuds gérés sur lesquels vous souhaitez exécuter une opération. Après avoir spécifié que vous souhaitez choisir les nœuds manuellement, le système affiche alors une liste de nœuds gérés sur lesquels vous pouvez exécuter l'opération.

Cette rubrique fournit des informations qui vous aideront à déterminer pourquoi un nœud géré confirmé comme en cours d'exécution ne figure pas dans vos listes de nœuds gérés dans Systems Manager.

Pour qu'un nœud soit géré par Systems Manager et figure dans les listes de nœuds gérés, il doit remplir trois conditions :

- SSM Agent doit être installé et exécuté sur le nœud avec un système d'exploitation pris en charge.

> **Note**
>
> Certains AWS managed Amazon Machine Images (AMIs) sont configurés pour lancer des instances SSM Agentpréinstallées. (Vous pouvez également configurer une AMI personnalisée pour préinstaller SSM Agent.) Pour plus d'informations, consultez Rechercher AMIs avec le SSM Agent préinstallé.

- Pour les instances Amazon Elastic Compute Cloud (Amazon EC2), vous devez associer AWS Identity and Access Management un profil d'instance (IAM) à l'instance. Ce profil d'instance permet à celles-ci de communiquer avec le service Systems Manager. Si vous n'attribuez pas de profil d'instance à l'instance, vous devez l'enregistrer à l'aide d'une activation hybride, ce qui ne constitue pas le scénario le plus fréquent.
- SSM Agent doit pouvoir se connecter à un point de terminaison Systems Manager afin de s'enregistrer auprès du service. Le nœud géré est ensuite disponible pour le service, comme le confirme le signal que celui-ci envoie toutes les cinq minutes afin de vérifier l'état de l'instance.
- Une fois que le statut d'un nœud géré a été `Connection Lost` pendant au moins 30 jours, il est possible que le nœud ne soit plus répertorié dans la console Fleet Manager. Pour le rétablir dans la liste, le problème à l'origine de la perte de connexion doit être résolu.

Après avoir vérifié qu'un nœud géré est bien en cours d'exécution, vous pouvez utiliser la commande suivante pour vous assurer que SSM Agent s'est enregistré avec succès auprès du service Systems Manager. Cette commande ne renvoie pas de résultats tant que l'enregistrement n'a pas réussi.

Linux & macOS

```
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 abouti et que le nœud géré est disponible pour les opérations de Systems Manager, la commande renvoie des résultats semblables aux suivants.

```
{
    "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 terminé ou a échoué, la commande renvoie des résultats semblables aux suivants :

```
{
    "InstanceAssociationStatusInfos": []
}
```

Si la commande ne renvoie aucun résultat au bout de 5 minutes environ, utilisez les informations suivantes pour résoudre les problèmes liés à vos nœuds gérés.

Rubriques

- Solution 1 : vérifier que SSM Agent est installé et en cours d'exécution sur le nœud géré
- Solution 2 : vérifier qu'un profil d'instance IAM a été spécifié pour l'instance (instances EC2 uniquement)
- Solution 3 : vérifier la connectivité des points de terminaison de service
- Solution 4 : vérifier la prise en charge du système d'exploitation cible
- Solution 5 : vérifier que vous travaillez de la même manière Région AWS que l'instance Amazon EC2
- Solution 6 : vérifier la configuration de proxy que vous avez appliquée à l'SSM Agent sur votre nœud géré

- Solution 7 : installer un certificat TLS sur les instances gérées
- Résolution des problèmes de disponibilité des nœuds gérés en utilisant ssm-cli

Solution 1 : vérifier que SSM Agent est installé et en cours d'exécution sur le nœud géré

Vérifiez que la dernière version de SSM Agent est installée et exécutée sur le nœud géré.

Pour déterminer si SSM Agent est installé et en cours d'exécution sur un nœud géré, consultez Vérification du statut de l'SSM Agent et démarrage de l'agent.

Pour installer ou réinstaller SSM Agent sur un nœud géré, consultez les rubriques suivantes :

- Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Linux
- Comment installer le SSM Agent sur des nœuds Linux hybrides
- Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Windows Server
- Comment installer le SSM Agent sur des nœuds Windows hybrides

Solution 2 : vérifier qu'un profil d'instance IAM a été spécifié pour l'instance (instances EC2 uniquement)

Vérifiez que l'instance Amazon Elastic Compute Cloud (Amazon EC2) est configurée avec un profil d'instance AWS Identity and Access Management (IAM) qui lui permet de communiquer avec l'API Systems Manager. Vérifiez également que votre utilisateur est associé à une politique d'approbation IAM qui permet à votre utilisateur afin de communiquer avec l'API Systems Manager.

> **Note**
>
> Les serveurs sur site, les appareils de périphérie et les machines virtuelles utilisent une fonction de service IAM plutôt qu'un profil d'instance. Pour plus d'informations, voir Créer le rôle de service IAM requis pour Systems Manager dans les environnements hybrides et multicloud.

Pour déterminer si un profil d'instance disposant des autorisations nécessaires est attaché à une instance EC2

1. Ouvrez la console Amazon EC2 à l'adresse https://console.aws.amazon.com/ec2/.
2. Dans le panneau de navigation, sélectionnez Instances.
3. Sélectionnez l'instance pour laquelle rechercher un profil d'instance.
4. Sous l'onglet Description du panneau inférieur, recherchez Rôle IAM et sélectionnez le nom du rôle.
5. Sur la page Résumé du rôle pour le profil d'instance, sous l'onglet Autorisations, vérifiez que `AmazonSSMManagedInstanceCore` est bien répertorié sous Politiques d'autorisations.

   Si vous utilisez plutôt une politique personnalisée, vérifiez qu'elle fournit les mêmes autorisations que `AmazonSSMManagedInstanceCore`.

   Ouvrir `AmazonSSMManagedInstanceCore` dans la console

   Pour plus d'informations sur les autres politiques qui peuvent être associées à un profil d'instance pour Systems Manager, voir Configurer les autorisations d'instance requises pour Systems Manager.

Solution 3 : vérifier la connectivité des points de terminaison de service

Vérifiez que l'instance est connectée aux points de terminaison de service Systems Manager. 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.

Pour les instances Amazon EC2, le point de terminaison du service Systems Manager de Région AWS l'instance est utilisé pour enregistrer l'instance si votre configuration de cloud privé virtuel (VPC) autorise le trafic sortant. Toutefois, si la configuration VPC avec laquelle l'instance a été lancée n'autorise pas le trafic sortant et que vous ne pouvez pas modifier cette configuration pour permettre la connexion aux points de terminaison de service publics, vous devez configurer des points de terminaison d'interface pour votre VPC.

Pour plus d'informations, consultez Améliorer la sécurité des instances EC2 en utilisant des points de terminaison VPC pour Systems Manager.

Solution 4 : vérifier la prise en charge du système d'exploitation cible

Vérifiez que l'opération que vous avez choisie peut être exécutée sur le type de nœud géré que vous souhaitez voir figurer dans la liste. Certaines opérations Systems Manager peuvent cibler uniquement des instances Windows ou uniquement des instances Linux. Par exemple, les documents Systems Manager (SSM) `AWS-InstallPowerShellModule` et `AWS-ConfigureCloudWatch`ne peuvent être exécutés que sur des instances Windows. Sur la page Exécuter une commande, si vous sélectionnez l'un de ces documents et que vous sélectionnez Choisir des instances manuellement, seules vos instances Windows sont répertoriées et disponibles à la sélection.

Solution 5 : vérifier que vous travaillez de la même manière Région AWS que l'instance Amazon EC2

Les instances Amazon EC2 sont créées et disponibles dans des régions spécifiques Régions AWS, telles que la région USA Est (Ohio) (us-east-2) ou Europe (Irlande) (eu-west-1). Assurez-vous que vous travaillez de la même manière Région AWS que l'instance Amazon EC2 avec laquelle vous souhaitez travailler. Pour de plus amples informations, consultez Choisir une région dans Mise en route avec la AWS Management Console.

Solution 6 : vérifier la configuration de proxy que vous avez appliquée à l'SSM Agent sur votre nœud géré

Vérifiez que la configuration de proxy que vous avez appliquée à l'SSM Agent sur votre nœud géré est correcte. Si la configuration de proxy est incorrecte, le nœud ne peut pas se connecter aux points de terminaison de service requis, ou Systems Manager risque de mal identifier le système d'exploitation du nœud géré. Pour plus d'informations, consultez Configuration SSM Agent pour utiliser un proxy sur les nœuds Linux et Configurer l'SSM Agent pour utiliser un proxy pour les instances Windows Server.

Solution 7 : installer un certificat TLS sur les instances gérées

Un certificat TLS (Transport Layer Security) doit être installé sur chaque instance gérée que vous utilisez. AWS Systems Manager Services AWS utilisez ces certificats pour chiffrer les appels vers d'autres Services AWS personnes.

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

Pour déterminer si le certificat requis est installé sur votre instance, exécutez la commande suivante en fonction du système d'exploitation de celle-ci. Assurez-vous de remplacer la partie *régionale* de l'URL par l' Région AWS emplacement de votre instance gérée.

Linux & macOS

```
curl -L https://ssm.region.amazonaws.com
```

Windows

```
Invoke-WebRequest -Uri https://ssm.region.amazonaws.com
```

La commande doit renvoyer une erreur `UnknownOperationException`. Si vous recevez un message d'erreur SSL/TLS, cela peut signifier que le certificat requis n'est pas installé.

Si vous constatez que les certificats Amazon Trust Services CA requis ne sont pas installés sur vos systèmes d'exploitation de base, sur les instances créées à partir de celles AMIs qui ne sont pas fournies par Amazon, ou sur vos propres serveurs et machines virtuelles sur site, vous devez installer et autoriser un certificat 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 plus d'informations sur ACM, consultez le Guide de l'utilisateur AWS Certificate Manager.

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

Pour plus d'informations sur les certificats Amazon Root et Starfield, consultez le billet de blog How to Prepare for AWS's Move to Its Own Certificate Authority.

Résolution des problèmes de disponibilité des nœuds gérés en utilisant **`ssm-cli`**

`ssm-cli` est un outil de ligne de commande autonome inclus dans l'installation SSM Agent. Lorsque vous installez la version SSM Agent 3.1.501.0 ou une version ultérieure sur une machine, vous pouvez exécuter des `ssm-cli` commandes sur cette machine. La sortie de ces commandes vous aide à déterminer si la machine répond aux exigences minimales requises pour être gérée par une instance Amazon EC2 ou une machine autre qu'EC2 AWS Systems Manager, et si elle est donc ajoutée aux listes de nœuds gérés dans Systems Manager. (SSM Agentla version 3.1.501.0 a été publiée en novembre 2021.)

Configuration requise

Pour qu'une instance Amazon EC2 ou une machine non EC2 soit gérée par AWS Systems Manager et disponible dans des listes de nœuds gérés, elle doit répondre à trois exigences principales :

- SSM Agent doit être installé et exécuté sur une machine dotée d'un système d'exploitation compatible.

  Certains AWS managed Amazon Machine Images (AMIs) pour EC2 sont configurés pour lancer des instances SSM Agentpréinstallées. (Vous pouvez également configurer une AMI personnalisée pour préinstaller SSM Agent.) Pour plus d'informations, consultez Rechercher AMIs avec le SSM Agent préinstallé.
- Un profil d'instance AWS Identity and Access Management (IAM) (pour les instances EC2) ou un rôle de service IAM (pour les machines non EC2) fournissant les autorisations requises pour communiquer avec le service Systems Manager doit être attaché à la machine.
- SSM Agent doit pouvoir se connecter à un point de terminaison Systems Manager afin de s'enregistrer auprès du service. Le nœud géré est ensuite disponible pour le service, comme le confirme le signal que celui-ci envoie toutes les cinq minutes afin de vérifier l'état du nœud géré.

Commandes préconfigurées dans **`ssm-cli`**

Il contient des commandes préconfigurées qui rassemblent les informations requises afin de déterminer pourquoi une machine, confirmée comme en cours d'exécution, ne figure pas dans vos listes de nœuds gérées dans Systems Manager. Ces commandes sont exécutées lorsque vous spécifiez l'option `get-diagnostics`.

Sur la machine, exécutez la commande suivante afin d'utiliser `ssm-cli` pour résoudre les problèmes de disponibilité des nœuds gérés.

Linux & macOS

```
ssm-cli get-diagnostics --output table
```

Windows

Sur des machines Windows Server, vous devez accéder au répertoire `C:\Program Files\Amazon\SSM` avant d'exécuter la commande.

```
ssm-cli.exe get-diagnostics --output table
```

PowerShell

Sur des machines Windows Server, vous devez accéder au répertoire `C:\Program Files\Amazon\SSM` avant d'exécuter la commande.

```
.\ssm-cli.exe get-diagnostics --output table
```

La sortie générée lors de l'exécution de cette commande renvoie un tableau similaire à celui qui suit.

> **Note**
>
> Les contrôles de connectivité vers les `monitoring` points de terminaison `ssmmessages` `s3` `kmslogs`,,, et concernent des fonctionnalités facultatives supplémentaires, telles Session Manager que la possibilité de se connecter à Amazon Simple Storage Service (Amazon S3) ou CloudWatch Amazon Logs, AWS Key Management Service et d'AWS KMS utiliser le chiffrement ().

Linux & macOS

```
[root@instance]# ssm-cli get-diagnostics --output table
##################################################################################################
# Check                                  # Status  # Note
                                       #
##################################################################################################
# EC2 IMDS                               # Success # IMDS is accessible and has
 instance id i-0123456789abcdefa in Region  #
#                                        #         # us-east-2
                                       #
```

```
##########################################################################################
# Hybrid instance registration          # Skipped # Instance does not have hybrid
 registration                           #
##########################################################################################
# Connectivity to ssm endpoint          # Success # ssm.us-east-2.amazonaws.com is
 reachable                              #
##########################################################################################
# Connectivity to ec2messages endpoint  # Success # ec2messages.us-
east-2.amazonaws.com is reachable                    #
##########################################################################################
# Connectivity to ssmmessages endpoint  # Success # ssmmessages.us-
east-2.amazonaws.com is reachable                    #
##########################################################################################
# Connectivity to s3 endpoint           # Success # s3.us-east-2.amazonaws.com is
 reachable                              #
##########################################################################################
# Connectivity to kms endpoint          # Success # kms.us-east-2.amazonaws.com is
 reachable                              #
##########################################################################################
# Connectivity to logs endpoint         # Success # logs.us-east-2.amazonaws.com is
 reachable                             #
##########################################################################################
# Connectivity to monitoring endpoint   # Success # monitoring.us-
east-2.amazonaws.com is reachable                     #
##########################################################################################
# AWS Credentials                       # Success # Credentials are for
                                       #
#                                       #         #
 arn:aws:sts::123456789012:assumed-role/Fullaccess/i-0123456789abcdefa #
#                                       #         # and will expire at 2021-08-17
 18:47:49 +0000 UTC                     #
##########################################################################################
# Agent service                         # Success # Agent service is running and is
 running as expected user              #
##########################################################################################
# Proxy configuration                   # Skipped # No proxy configuration detected
                                       #
##########################################################################################
# SSM Agent version                     # Success # SSM Agent version is 3.0.1209.0,
 latest available agent version is     #
#                                       #         # 3.1.192.0
                                       #
##########################################################################################
```

Windows Server and PowerShell

```
PS C:\Program Files\Amazon\SSM> .\ssm-cli.exe get-diagnostics --output table
###############################################################################################
# Check                                  # Status  # Note
                                       #
###############################################################################################
# EC2 IMDS                               # Success # IMDS is accessible and has
 instance id i-0123456789EXAMPLE in       #
#                                        #         # Region us-east-2
                                       #
###############################################################################################
# Hybrid instance registration           # Skipped # Instance does not have hybrid
 registration                            #
###############################################################################################
# Connectivity to ssm endpoint           # Success # ssm.us-east-2.amazonaws.com is
 reachable                              #
###############################################################################################
# Connectivity to ec2messages endpoint  # Success # ec2messages.us-
east-2.amazonaws.com is reachable                   #
###############################################################################################
# Connectivity to ssmmessages endpoint  # Success # ssmmessages.us-
east-2.amazonaws.com is reachable                   #
###############################################################################################
# Connectivity to s3 endpoint            # Success # s3.us-east-2.amazonaws.com is
 reachable                              #
###############################################################################################
# Connectivity to kms endpoint           # Success # kms.us-east-2.amazonaws.com is
 reachable                             #
###############################################################################################
# Connectivity to logs endpoint          # Success # logs.us-east-2.amazonaws.com is
 reachable                             #
###############################################################################################
# Connectivity to monitoring endpoint   # Success # monitoring.us-
east-2.amazonaws.com is reachable                    #
###############################################################################################
# AWS Credentials                        # Success # Credentials are for
                                       #
#                                        #         #
 arn:aws:sts::123456789012:assumed-role/SSM-Role/i-123abc45EXAMPLE  #
#                                        #         #  and will expire at 2021-09-02
 13:24:42 +0000 UTC                    #
###############################################################################################
```

```
# Agent service                          # Success # Agent service is running and is
 running as expected user            #
##########################################################################################
# Proxy configuration                    # Skipped # No proxy configuration detected
                                      #
##########################################################################################
# Windows sysprep image state            # Success # Windows image state value is at
 desired value IMAGE_STATE_COMPLETE  #
##########################################################################################
# SSM Agent version                      # Success # SSM Agent version is 3.2.815.0,
 latest agent version in us-east-2   #
#                                       #         # is 3.2.985.0
                                      #
##########################################################################################
```

Le tableau suivant fournit des détails supplémentaires pour chacune des vérifications effectuées par `ssm-cli`.

Vérifications de diagnostic **`ssm-cli`**

Check	Détails
Service de métadonnées d'instance Amazon EC2	Indique si le nœud géré est en mesure d'accéder au service de métadonnées. Un test qui échoue indique un problème de connectivité à `http://169.254.169.254` qui peut être dû à la configuration de l'acheminement local, du proxy, ou du pare-feu et du proxy du système d'exploitation (OS).
Enregistrement d'une instance hybride	Indique si SSM Agent est enregistré à l'aide d'une activation hybride.
Connectivité au point de terminaison `ssm`	Indique si le nœud a accès aux points de terminaison de service de Systems Manager sur le port TCP 443. Un test raté indique des problèmes de connectivité `https://ssm.` *`region`*`.amazonaws.com` en fonction de l' Région AWS emplacement du nœud. Les problèmes de connectivité peuvent être dus

Check	Détails
	à la configuration du VPC, et notamment aux groupes de sécurité, aux listes de contrôle d'accès réseau, aux tables de routage ou aux pare-feu et proxies du système d'exploitation.
Connectivité au point de terminaison `ec2messages`	Indique si le nœud a accès aux points de terminaison de service de Systems Manager sur le port TCP 443. Un test raté indique des problèmes de connectivité `https://ec2messages.` *`region`*`.amazonaws.com` en fonction de l' Région AWS emplacement du nœud. Les problèmes de connectivité peuvent être dus à la configuration du VPC, et notamment aux groupes de sécurité, aux listes de contrôle d'accès réseau, aux tables de routage ou aux pare-feu et proxies du système d'exploitation.
Connectivité au point de terminaison `ssmmessages`	Indique si le nœud a accès aux points de terminaison de service de Systems Manager sur le port TCP 443. Un test raté indique des problèmes de connectivité `https://ssmmessages.` *`region`*`.amazonaws.com` en fonction de l' Région AWS emplacement du nœud. Les problèmes de connectivité peuvent être dus à la configuration du VPC, et notamment aux groupes de sécurité, aux listes de contrôle d'accès réseau, aux tables de routage ou aux pare-feu et proxies du système d'exploitation.

Check	Détails
Connectivité au point de terminaison `s3`	Indique si le nœud est en mesure d'atteind re le point de terminaison d'Amazon Simple Storage Service sur le port TCP 443. Un test raté indique des problèmes de connectivité `https://s3.` *`region`*`.amazonaws.com` en fonction de l' Région AWS emplacement du nœud. La connectivité à ce point de terminais on n'est pas nécessaire pour qu'un nœud apparaisse dans votre liste de nœuds gérés.
Connectivité au point de terminaison `kms`	Indique si le nœud est capable d'atteindre le point de terminaison du service AWS Key Management Service sur le port TCP 443. Un test raté indique des problèmes de connectivité `https://kms.` *`region`*`.amazonaws.com` en fonction de l' Région AWS emplacement du nœud. La connectivité à ce point de terminais on n'est pas nécessaire pour qu'un nœud apparaisse dans votre liste de nœuds gérés.
Connectivité au point de terminaison `logs`	Indique si le nœud est capable d'atteindre le point de terminaison du service pour Amazon CloudWatch Logs sur le port TCP 443. Un test raté indique des problèmes de connectivité `https://logs.` *`region`*`.amazonaws.com` en fonction de l' Région AWS emplacement du nœud. La connectivité à ce point de terminais on n'est pas nécessaire pour qu'un nœud apparaisse dans votre liste de nœuds gérés.

Check	Détails
Connectivité au point de terminaison `monitoring`	Indique si le nœud est capable d'atteind re le point de terminaison du service pour Amazon CloudWatch sur le port TCP 443. Un test raté indique des problèmes de connectivité `https://monitoring . region.amazonaws.com` en fonction de l' Région AWS emplacement du nœud. La connectivité à ce point de terminaison n'est pas nécessaire pour qu'un nœud apparaisse dans votre liste de nœuds gérés.
AWS Informations d'identification	Indique si l'SSM Agent dispose des informati ons d'identification requises en fonction du profil d'instance IAM (pour les instances EC2) ou de la fonction du service IAM (pour les machines non EC2) attaché à la machine. Un test qui échoue indique qu'aucun profil d'instance IAM ou fonction du service IAM n'est attaché à la machine, ou qu'il ne contient pas les autorisations requises pour Systems Manager.
Service d'agent	Indique si le service SSM Agent est en cours d'exécution, et s'il est exécuté en tant que racine pour Linux ou macOS, ou SYSTEM pour Windows Server. Un test qui échoue indique que le service SSM Agent n'est pas en cours d'exécution ou qu'il n'est pas exécuté en tant que root ou SYSTEM.
Configuration du proxy	Indique si SSM Agent est configuré pour utiliser un proxy.

Check	Détails
État de l'image Sysprep (Windows uniquement)	Indique l'état de `Sysprep` sur le nœud. SSM Agent ne démarrera pas sur le nœud si l'état de `Sysprep` est une valeur autre que `IMAGE_STATE_COMPLETE` .
Version de SSM Agent	Indique si la dernière version disponible de l'SSM Agent est installée.

Conformité d'AWS Systems Manager

Vous pouvez utiliser Compliance, une fonctionnalité de AWS Systems Manager, pour analyser votre parc de nœuds gérés afin de détecter la conformité des correctifs et les incohérences de configuration. Vous pouvez collecter et agréger des données provenant de plusieurs Comptes AWS régions, puis explorer les ressources spécifiques qui ne sont pas conformes. Par défaut, le service Conformité affiche les données de conformité actuelles relatives à l'application de correctifs dans Patch Manager et aux associations dans State Manager. (Patch Manager et State Manager sont également des fonctionnalités de AWS Systems Manager.) Pour vos débuts dans la section conformité, ouvrez Systems Manager console (la console Systems Manager). Dans le panneau de navigation, sélectionnez Compliance (Conformité).

Les données de conformité des correctifs Patch Manager peuvent être envoyées à AWS Security Hub. Security Hub vous offre une vue complète sur vos alertes de sécurité haute priorité et votre statut de conformité. Il surveille également le statut d'application des correctifs de votre flotte. Pour plus d'informations, consultez Intégration Patch Manager avec AWS Security Hub.

La conformité offre les avantages et fonctions supplémentaires suivants :

- Affichage du suivi des modifications et de l'historique de conformité pour les données de correctifs de Patch Manager et les associations State Manager à l'aide d' AWS Config.
- Personnalisation de la conformité pour créer vos propres types de conformité en fonction de vos exigences métier ou informatiques.
- Résolvez les problèmes en utilisant Run Command une autre fonctionnalité de AWS Systems ManagerState Manager, ou Amazon EventBridge.
- Transférez les données vers Amazon Athena et Amazon QuickSight pour générer des rapports à l'échelle de la flotte.

EventBridge soutien

Cette fonctionnalité de Systems Manager est prise en charge en tant que type d'événement dans les EventBridge règles d'Amazon. Pour plus d'informations, consultez Surveillance d'événements Systems Manager avec Amazon EventBridge et Référence : modèles et types d'événements Amazon EventBridge pour Systems Manager.

intégration d'Chef InSpec

Systems Manager s'intègre à Chef InSpec. InSpec est un framework d'exécution open source qui vous permet de créer des profils lisibles par l'homme sur GitHub Amazon Simple Storage Service (Amazon S3). Vous pouvez ensuite utiliser Systems Manager pour exécuter des analyses de conformité et afficher les nœuds gérés conformes et non conformes. Pour plus d'informations, consultez Utilisation de Chef InSpec profils avec Systems Manager Compliance.

Tarification

Le service Compliance est fourni sans frais supplémentaires. Vous ne payez que pour les AWS ressources que vous utilisez.

Table des matières

Mise en route avec le service Conformité

Pour commencer à utiliser le service Compliance, une fonctionnalité de AWS Systems Manager, effectuez les tâches suivantes.

Tâche	Pour plus d'informations
Le service Compliance utilise des données d'application de correctifs dans Patch Manager	Con AWS Systems Manager figuration

Tâche	Pour plus d'informations
et des associations dans State Manager. (Patch Manager et State Manager sont également des fonctionnalités de AWS Systems Manager.) Le service Conformité utilise également des types de conformité personnalisés sur les nœud gérés avec Systems Manager. Vérifiez que vous avez satisfait la configuration requise pour vos instances Amazon Elastic Compute Cloud (Amazon EC2) et vos machines non EC2 dans un environnement hybride et multicloud.	
Mettez à jour Systems Manager SSM Agent (SSM Agent) sur vos nœuds gérés avec la dernière version.	Utilisation de l'option SSM Agent
Si vous prévoyez de surveiller la conformité des correctifs, vérifiez que vous avez configuré Patch Manager. Vous devez effectuer des opérations d'application de correctifs à l'aide de Patch Manager pour que le service Compliance puisse afficher les données de conformité des correctifs.	AWS Systems Manager Patch Manager
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 Compliance puisse afficher les données de conformité des associations.	AWS Systems Manager State Manager
(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

Tâche	Pour plus d'informations
(Facultatif) Créez des types de conformité personnalisée.	Démonstration du service Conformité (AWS CLI)
(Facultatif) Créez une synchronisation de données de ressources pour agréger toutes les données de conformité dans un compartiment Amazon Simple Storage Service (Amazon S3) cible.	Création d'une synchronisation de données de ressources pour Compliance

Création d'une synchronisation de données de ressources pour Compliance

Vous pouvez utiliser la fonctionnalité de synchronisation des données de ressources AWS Systems Manager pour envoyer les données de conformité de tous vos nœuds gérés vers un compartiment Amazon Simple Storage Service (Amazon S3) cible. Lorsque vous créez la synchronisation, vous pouvez spécifier des nœuds gérés à partir de plusieurs Comptes AWS, Régions AWS, ainsi que de votre environnement hybride et multicloud. La synchronisation de données de ressources met alors automatiquement à jour les données centralisées lors de la collecte de nouvelles données de conformité. Toutes les données de conformité étant stockées dans un compartiment S3 cible, vous pouvez utiliser des services tels qu'Amazon Athena et Amazon QuickSight pour interroger et analyser les données agrégées. La configuration de la synchronisation de données de ressources pour Compliance est une opération unique.

Utilisez la procédure suivante pour créer une synchronisation de données de ressources pour Compliance en utilisant la AWS Management Console.

Pour créer et configurer un compartiment S3 pour la synchronisation de données de ressources (console)

1. Ouvrez la console Amazon S3 sur https://console.aws.amazon.com/s3/.
2. Créez un compartiment pour stocker vos données de conformité agrégées. Pour plus d'informations, consultez Création d'un compartiment dans le Guide de l'utilisateur d'Amazon Simple Storage Service. Notez le nom du bucket et l' Région AWS endroit où vous l'avez créé.
3. Ouvrez le compartiment, sélectionnez l'onglet Autorisations, puis sélectionnez Politique de compartiment.

4. Copiez et collez la politique de compartiment suivante dans l'éditeur de politique. Remplacez DOC-EXAMPLE-BUCKET et *Account-ID* par le nom du compartiment S3 que vous avez créé et un identifiant valide. Compte AWS Vous avez également la possibilité de remplacer *Bucket-Prefix* par le nom d'un préfixe Amazon S3 (sous-répertoire). Si vous n'avez pas créé de préfixe, supprimez *Bucket-Prefix*/ de l'ARN dans la politique.

```
{
    "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 AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.

3. Sélectionnez Accoung management (Gestion du compte), Resource Data Syncs (Synchronisations de données de ressources), puis Create resource data sync (Créer une synchronisation de données de ressources).
4. Dans le champ Sync name (Nom de la synchronisation), saisissez un nom pour la configuration de la synchronisation.
5. Dans le champ Bucket name (Nom du compartiment), saisissez le nom du compartiment Amazon S3 créé au début de cette procédure.
6. (Facultatif) Dans le champ Préfixe du compartiment, saisissez le nom d'un préfixe de compartiment S3 (sous-répertoire).
7. Dans le champ Région du compartiment, sélectionnez Cette région si le compartiment S3 créé est localisé dans la Région AWS actuelle. Si le compartiment se trouve dans une autre région Région AWS, choisissez Autre région et entrez 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 plus d'informations, consultez Tarification Amazon S3.

8. Sélectionnez Create (Créer).

Utilisation du service Conformité

Conformité, une capacité de AWS Systems Manager collecte et de rapport des données sur l'état de l'application des correctifs lors de l'application de Patch Manager correctifs et des associations dans. State Manager (Patch Manageret State Manager sont également toutes deux des fonctionnalités de AWS Systems Manager.) Compliance rapporte également les types de conformité personnalisés que vous avez spécifiés pour vos nœuds gérés. 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 Manager s'intègre à Chef InSpec. InSpec est un framework d'exécution open source qui vous permet de créer des profils lisibles par l'homme sur GitHub Amazon Simple

Storage Service (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 Utilisation de Chef InSpec profils avec Systems Manager Compliance.

A propos de la conformité des correctifs

Une fois que vous avez utilisé Patch Manager pour installer des correctifs sur vos instances, les informations relatives au statut de conformité sont immédiatement disponibles dans la console, ou en réponse à des commandes de l' AWS Command Line Interface (AWS CLI) ou aux opérations d'API Systems Manager correspondantes.

Pour de plus amples informations sur les valeurs du statut de conformité des correctifs, veuillez consulter Comprendre les valeurs d'état de conformité des correctifs.

A propos de la conformité des associations State Manager

Une fois que vous avez créé une ou plusieurs State Manager associations, les informations relatives au statut de conformité sont immédiatement disponibles dans la console ou en réponse aux AWS CLI commandes ou aux opérations d'API Systems Manager correspondantes. Pour les associations, le service Compliance affiche les statuts `Compliant` ou `Non-compliant` et le niveau de sévérité attribué à l'association, tel que `Critical` ou `Medium`.

A propos de la conformité personnalisée

Vous pouvez attribuer des métadonnées de conformité à un nœud géré. 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 nœuds gérés. 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'opération PutComplianceItemsAPI pour indiquer explicitement quels nœuds gérés exécutent d'anciennes versions du logiciel X. Vous ne pouvez attribuer des métadonnées de conformité qu'à l' AWS CLI aide du ou des SDK. AWS Tools for Windows PowerShell L'exemple suivant de commande d'interface de ligne de commande attribue des métadonnées de conformité à une instance gérée et spécifie le type de conformité au format requis `Custom:`. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations.

Linux & macOS

```
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
```

> **Note**
>
> Le paramètre `ResourceType` prend uniquement en charge `ManagedInstance`. Si vous ajoutez une conformité personnalisée à un appareil principal AWS IoT Greengrass géré, vous devez spécifier un `ResourceType` de `ManagedInstance`.

Les gestionnaires de la conformité peuvent ensuite afficher des récapitulatifs ou créer des rapports sur les nœuds gérés conformes ou non. Vous pouvez attribuer au maximum 10 types différents de conformité personnalisée à un nœud géré.

Pour obtenir un exemple montrant comment créer un type de conformité personnalisée et afficher les données de conformité, consultez Démonstration du service Conformité (AWS CLI).

Affichage des données de conformité actuelles

Cette section explique comment afficher les données de conformité dans la console Systems Manager et à l'aide de la 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.

Rubriques

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 AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Compliance (Conformité).
3. Dans la section Compliance dashboard filtering (Filtrage du tableau de bord de conformité), sélectionnez une option pour filtrer les données de conformité. La section Compliance resources summary (Synthèse des ressources de conformité) affiche le nombre de données de conformité en fonction du filtre que vous avez choisi.
4. Pour explorer une ressource en détail, faites défiler vers le bas jusqu'à la zone Details overview for resources (Vue d'ensemble détaillée des ressources) et sélectionnez l'ID d'un nœud géré.
5. Sur la page Instance ID (ID d'instance) ou Name (Nom), sélectionnez l'onglet Configuration compliance (Conformité de la configuration) afin d'afficher un rapport détaillé de conformité de la configuration pour le nœud géré.

> **Note**
>
> Pour de plus amples informations sur la résolution des problèmes de conformité, veuillez consulter Résolution des problèmes de conformité avec EventBridge.

Affichage des données de conformité actuelles (AWS CLI)

Vous pouvez consulter les résumés des données de conformité pour les correctifs, les associations et les types de conformité personnalisés dans le in à l'aide AWS CLI des commandes suivantes AWS CLI .

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 sévérité 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 une illustration de la configuration des correctifs et pour consulter les détails de conformité des correctifs à l'aide du AWS CLI, voirDidacticiel : application de correctifs à un environnement de serveur (AWS CLI).

Affichage du suivi des modifications et de l'historique de configuration de la conformité

Le service Systems Manager Compliance affiche les données actuelles de conformité des associations et de l'application de correctifs pour vos nœuds gérés. Vous pouvez consulter l'historique de conformité des correctifs et des associations ainsi que le suivi des modifications en utilisant AWS Config. AWS Config fournit une vue détaillée de la configuration des AWS ressources de 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 Guide du développeur AWS Config .

> **Note**
>
> Pour plus d'informations sur la AWS Config tarification, consultez la section Tarification.

Suppression d'une synchronisation de données de ressources pour le service Conformité

Si vous ne souhaitez plus utiliser AWS Systems Manager Compliance pour consulter les données de conformité, nous vous recommandons également de supprimer les synchronisations des données de ressources utilisées pour la collecte des données de conformité.

Pour supprimer une synchronisation de données de ressources Compliance

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Sélectionnez Account management (Gestion de compte), Resource data syncs (Synchronisation de données de ressources).
4. Dans la liste, sélectionnez une synchronisation.

> **Important**
>
> Veillez à bien choisir la synchronisation utilisée pour le service Conformité. Systems Manager prend en charge la synchronisation de données de ressources pour plusieurs

fonctionnalités. Si vous choisissez la mauvaise synchronisation, vous risquez de perturber l'agrégation des données pour Systems Manager Explorer ou Systems Manager Inventory.

5. Sélectionnez Delete (Supprimer).
6. Supprimez le compartiment Amazon Simple Storage Service (Amazon S3) dans lequel les données ont été enregistrées. Pour obtenir des informations sur la suppression d'un compartiment Amazon S3, consultez Deleting a bucket (Suppression d'un compartiment).

Résolution des problèmes de conformité avec EventBridge

Vous pouvez corriger rapidement les problèmes de conformité des correctifs et des associations en utilisant la fonctionnalité Run Command, une fonctionnalité de AWS Systems Manager. Vous pouvez cibler une instance ou les ID ou balises des appareils principaux AWS IoT Greengrass et exécutez le document `AWS-RunPatchBaseline` ou `AWS-RefreshAssociation`. Si actualiser l'association ou exécuter à nouveau le référentiel de correctif ne permet pas de résoudre le problème de conformité, vous devez vérifier vos associations, référentiels de correctifs ou configurations d'instance pour comprendre pourquoi les opérations Run Command n'ont pas permis de résoudre le problème.

Pour de plus amples informations sur l'application de correctifs, veuillez consulter AWS Systems Manager Patch Manager et À propos du document SSM `AWS-RunPatchBaseline`.

Pour de plus amples informations sur les associations, veuillez consulter Utilisation d'associations dans Systems Manager.

Pour de plus amples informations sur l'exécution d'une commande, veuillez consulter AWS Systems Manager Run Command.

Spécifier Compliance comme cible d'un événement EventBridge

Vous pouvez également configurer Amazon EventBridge pour effectuer une action en réponse à des événements Systems Manager Compliance. Par exemple, si un ou plusieurs nœuds gérés 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 document `AWS-RunPatchBaseline` ou `AWS-RefreshAssocation` lorsque l'événement Compliance se produit.

Utilisez la procédure suivante afin de configurer Compliance comme cible d'un événement EventBridge.

Pour configurer Compliance 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.
3. Choisissez Create rule.
4. 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 AWS et sur le même bus d'événement.
5. Pour Event bus (Bus d'événement), sélectionnez le bus d'événement que vous souhaitez associer à cette règle. Si vous souhaitez que cette règle s'applique aux événements correspondants provenant de votre propre Compte AWS, sélectionnez défaut. Lorsqu'un Service AWS de votre compte émet un événement, il accède toujours au bus d'événement par défaut de votre compte.
6. Pour Rule type (Type de règle), choisissez Rule with an event pattern (Règle avec un modèle d'événement).
7. Choisissez Next (Suivant).
8. Pour Event source (Origine de l'événement), choisissez events or EventBridge partner events (Événements AWS ou événements partenaires EventBridge).
9. Dans la section Event pattern (Modèle d'événement), choisissez Event pattern form (Modèle d'événement).
10. Pour Event source (Origine de l'événement), choisissez AWSservices (Services).
11. Pour le AWS service choisissez Systems Manager.
12. Dans le champ Event type (Type d'événement), sélectionnez Configuration Compliance (Conformité de configuration).
13. Pour Specific detail type(s) (Type(s) de détails spécifiques), choisissez Configuration Compliance State Change (Changements d'état de la conformité de configuration).
14. Choisissez Next (Suivant).
15. Pour Types de cibles, choisissez service AWS.
16. Pour Target (Cible), sélectionnez Systems Manager Run Command.
17. Dans la liste Document, sélectionnez un document Systems Manager (document SSM) à exécuter lorsque la cible sera invoquée. Par exemple, sélectionnez `AWS-RunPatchBaseline` pour un événement de correctif non conforme, ou `AWS-RefreshAssociation` pour un événement d'association non conforme.

18. 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.

19. Choisissez Next (Suivant).
20. (Facultatif) Saisissez une ou plusieurs balises pour la règle. Pour plus d'informations, consultez Balisage de vos ressources Amazon EventBridge dans le Guide de l'utilisateur Amazon EventBridge.
21. Choisissez Next (Suivant).
22. Consultez les détails de la règle et choisissez Create rule (Créer une règle).

Démonstration du service Conformité (AWS CLI)

La procédure suivante vous guide tout au long du processus d'utilisation de l'AWS Command Line Interface (AWS CLI) afin d'appeler l'opération d'API AWS Systems Manager PutComplianceItems pour attribuer des métadonnées de conformité personnalisée à une ressource. Vous pouvez aussi utiliser cette opération d'API pour attribuer manuellement des métadonnées de conformité des correctifs ou des associations à un nœud géré, comme indiqué dans la démonstration suivante. Pour de plus amples informations sur la conformité personnalisée, consultez A propos de la conformité personnalisée.

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 Command Line Interface (AWS CLI).

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l'AWS CLI.
2. Exécutez la commande suivante pour attribuer des métadonnées de conformité personnalisée à un nœud géré. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations. Le paramètre `ResourceType` ne prend en charge qu'une valeur de `ManagedInstance`. Spécifiez cette valeur même si vous attribuez des métadonnées de conformité personnalisée à un appareil principal AWS IoT Greengrass géré.

Linux & macOS

```
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 un ou plusieurs nœuds. Vous pouvez aussi utiliser les commandes suivantes pour attribuer manuellement des métadonnées de conformité des correctifs ou des associations à des nœuds gérés :

   Métadonnées de conformité des associations

   Linux & macOS

```
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 & macOS

```
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 un nœud géré spécifique. Utilisez des filtres pour explorer en détail des données de conformité spécifiques.

   Linux & macOS

   ```
   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 & macOS

   ```
   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 & macOS

```
aws ssm list-resource-compliance-summaries \
    --filters Key=OverallSeverity,Values=UNSPECIFIED
```

Windows

```
aws ssm list-resource-compliance-summaries ^
    --filters Key=OverallSeverity,Values=UNSPECIFIED
```

Linux & macOS

```
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 & macOS

```
aws ssm list-resource-compliance-summaries \
    --filters Key=ExecutionType,Values=Command
```

Windows

```
aws ssm list-resource-compliance-summaries ^
```

```
    --filters Key=ExecutionType,Values=Command
```

Linux & macOS

```
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 & macOS

```
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 & macOS

```
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
```

AWS Systems Manager Inventory

AWS Systems Manager Inventory vous permet d'avoir une certaine visibilité sur votre environnement informatique AWS. Vous pouvez utiliser l'inventaire pour collecter les métadonnées à partir de vos nœuds gérés. 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 quels nœuds exécutent les logiciels et les configurations requis par votre politique logicielle, et quels nœuds doivent être mis à jour. Vous pouvez configurer l'inventaire sur l'ensemble de vos nœuds gérés à l'aide d'une procédure en un clic. Vous pouvez également configurer et afficher les données d'inventaire de plusieurs Régions AWS et Comptes AWS. Pour vos premiers pas dans l'inventaire, ouvrez la console Systems Manager. Dans le volet de navigation, sélectionnez Inventory.

Si les types de métadonnées préconfigurés collectés par Systems Manager Inventory 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 au nœud géré dans un répertoire spécifique. Lorsque Systems Manager Inventory 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**
>
> Systems Manager Inventory collecte uniquement les métadonnées de vos nœuds gérés. L'inventaire n'accède pas aux données ou aux informations propriétaires.

Le tableau suivant décrit les types de données que vous pouvez collecter avec Systems Manager Inventory. Le tableau décrit également les différentes offres de ciblage des nœuds et les intervalles de collecte que vous pouvez spécifier.

Configuration	Détails
Types de métadonnées	Vous pouvez configurer l'inventaire pour collecter les types de données suivants : • Applications : noms des applications, éditeurs, versions, etc. • Composants AWS : pilote EC2, agents, versions, etc. • Fichiers : Nom, taille, version, date d'installation, heures de modification et du dernier accès, etc. • Configuration réseau : adresse IP, adresse MAC, DNS, passerelle, masque de sous-réseau, etc. • Mises à jour Windows : ID du correctif, auteur de l'installation, date d'installation, etc. • Détails de l'instance : nom du système, nom des systèmes d'exploitation, version du système d'exploitation, 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 affectées à vos nœuds. • Registre Windows : Chemin de la clé de registre, nom de valeur, type de valeur et valeur.

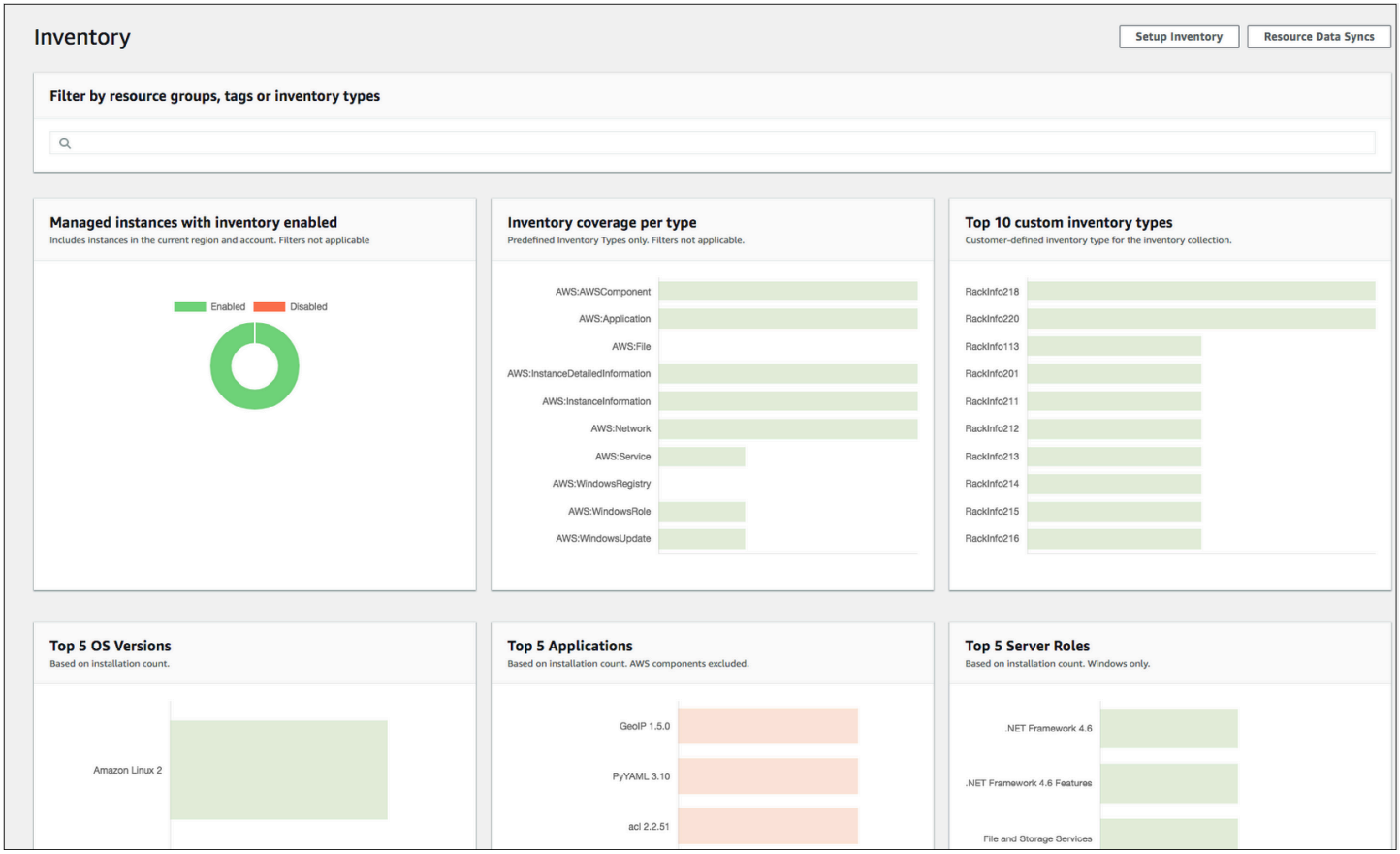
Configuration	Détails
	• Rôles Windows : Nom, nom d'affichage, chemin, type de fonction, état d'installation, etc. • Inventaire personnalisé : Métadonnées affectées à un nœud géré, comme décrit dans Utilisation de l'inventaire personnalisé. **Note** Pour afficher la liste de toutes les métadonnées collectées par l'inventaire, consultez Métadonnées collectées par inventaire.
Nœuds à cibler	Vous pouvez choisir d'inventorier tous les nœuds gérés dans votre Compte AWS, de sélectionner individuellement des nœuds ou de cibler des groupes de nœuds à l'aide de balises. Pour plus d'informations sur la collecte de données d'inventaire sur l'ensemble de vos nœuds gérés, consultez Répertoriez tous les nœuds gérés de votre Compte AWS.
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 données 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 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 nœuds gérés des appareils principaux sur site et AWS IoT Greengrass, les cartes d'inventaire filtrent automatiquement les nœuds présentant l'état Terminated (Résilié).

Si vous créez une synchronisation de données de ressources pour synchroniser et stocker toutes vos données dans un seul compartiment Amazon S3, vous pouvez explorer en détail les données sur la

page Inventory Detailed View (Vue détaillée d'inventaire). Pour de plus amples informations, veuillez consulter Interrogation des données d'inventaire à partir de plusieurs régions et comptes.

Prise en charge d'EventBridge

Cette fonctionnalité de Systems Manager est prise en charge en tant que type d'événement dans les règles Amazon EventBridge. Pour plus d'informations, consultez Surveillance d'événements Systems Manager avec Amazon EventBridge et Référence : modèles et types d'événements Amazon EventBridge pour Systems Manager.

Table des matières

En savoir plus sur Systems Manager Inventory

Lorsque vous configurez l'inventaire AWS Systems Manager, vous spécifiez le type de métadonnées à collecter, les nœuds à partir desquels les métadonnées doivent être collectées et un calendrier de collecte des métadonnées. Ces configurations sont enregistrées avec votre Compte AWS sous forme d'association AWS 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
- Utilisation de l'inventaire de fichiers et du registre Windows
- Services AWS connexe

Métadonnées collectées par inventaire

L'exemple suivant affiche la liste complète des métadonnées collectées par chaque plugin Inventory AWS Systems Manager.

```
{
    "typeName": "AWS:InstanceInformation",
    "version": "1.0",
    "attributes":[
      { "name": "AgentType",                                "dataType" : "STRING"},
      { "name": "AgentVersion",                             "dataType" : "STRING"},
      { "name": "ComputerName",                             "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"},
      { "name": "AgentStatus",                              "dataType" : "STRING"},
      { "name": "InstanceStatus",                           "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": "Release",           "dataType": "STRING"},
      { "name": "Epoch",             "dataType": "STRING"},
      { "name": "InstalledTime",     "dataType": "STRING"},
      { "name": "Architecture",      "dataType": "STRING"},
      { "name": "URL",               "dataType": "STRING"},
      { "name": "Summary",           "dataType": "STRING"},
      { "name": "PackageId",         "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:Process",
    "version": "1.0",
    "attributes":[
      { "name": "StartTime",          "dataType": "STRING"},
      { "name": "CommandLine",        "dataType": "STRING"},
      { "name": "User",               "dataType": "STRING"},
      { "name": "FileName",           "dataType": "STRING"},
      { "name": "FileVersion",        "dataType": "STRING"},
      { "name": "FileDescription",    "dataType": "STRING"},
      { "name": "FileSize",           "dataType": "STRING"},
      { "name": "CompanyName",        "dataType": "STRING"},
      { "name": "ProductName",        "dataType": "STRING"},
      { "name": "ProductVersion",     "dataType": "STRING"},
      { "name": "InstalledDate",      "dataType": "STRING"},
      { "name": "InstalledDir",       "dataType": "STRING"},
      { "name": "UsageId",            "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": "InstalledPendingRebootCount",   "dataType": "NUMBER"},
      { "name": "InstalledOtherCount",           "dataType": "NUMBER"},
      { "name": "InstalledRejectedCount",        "dataType": "NUMBER"},
      { "name": "NotApplicableCount",            "dataType": "NUMBER"},
```

```
      { "name": "UnreportedNotApplicableCount",     "dataType": "NUMBER"},
      { "name": "MissingCount",                     "dataType": "NUMBER"},
      { "name": "FailedCount",                      "dataType": "NUMBER"},
      { "name": "OperationType",                    "dataType": "STRING"},
      { "name": "OperationStartTime",               "dataType": "STRING"},
      { "name": "OperationEndTime",                 "dataType": "STRING"},
      { "name": "InstallOverrideList",              "dataType": "STRING"},
      { "name": "RebootOption",                     "dataType": "STRING"},
      { "name": "LastNoRebootInstallOperationTime", "dataType": "STRING"},
      { "name": "ExecutionId",                      "dataType": "STRING",
   "isOptional": "true"},
      { "name": "NonCompliantSeverity",             "dataType": "STRING",
   "isOptional": "true"},
      { "name": "SecurityNonCompliantCount",        "dataType": "NUMBER",
   "isOptional": "true"},
      { "name": "CriticalNonCompliantCount",        "dataType": "NUMBER",
   "isOptional": "true"},
      { "name": "OtherNonCompliantCount",           "dataType": "NUMBER",
   "isOptional": "true"}
    ]
  },
  {
    "typeName": "AWS:PatchCompliance",
    "version":"1.0",
    "attributes":[
      { "name": "Title",                       "dataType": "STRING"},
      { "name": "KBId",                        "dataType": "STRING"},
      { "name": "Classification",              "dataType": "STRING"},
      { "name": "Severity",                    "dataType": "STRING"},
      { "name": "State",                       "dataType": "STRING"},
      { "name": "InstalledTime",               "dataType": "STRING"}
    ]
  },
  {
    "typeName": "AWS:ComplianceItem",
    "version":"1.0",
    "attributes":[
      { "name": "ComplianceType",              "dataType": "STRING",
 "isContext": "true"},
      { "name": "ExecutionId",                 "dataType": "STRING",
 "isContext": "true"},
      { "name": "ExecutionType",               "dataType": "STRING",
 "isContext": "true"},
```

```
      { "name": "ExecutionTime",                "dataType": "STRING",
"isContext": "true"},
      { "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"},
      { "name": "InstallOverrideList",          "dataType": "STRING",
"isOptional": "true"},
      { "name": "DetailedText",                 "dataType": "STRING",
"isOptional": "true"},
      { "name": "DetailedLink",                 "dataType": "STRING",
"isOptional": "true"},
      { "name": "CVEIds",                       "dataType": "STRING",
"isOptional": "true"}
    ]
 },
 {
   "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"}
      ]
    },
    {
      "typeName": "AWS:Tag",
      "version":"1.0",
      "attributes":[
        { "name": "Key",                     "dataType": "STRING"},
        { "name": "Value",                   "dataType": "STRING"}
      ]
    },
    {
      "typeName": "AWS:ResourceGroup",
      "version":"1.0",
      "attributes":[
        { "name": "Name",                   "dataType": "STRING"},
        { "name": "Arn",                    "dataType": "STRING"}
      ]
    },
    {
      "typeName": "AWS:BillingInfo",
      "version": "1.0",
      "attributes": [
        { "name": "BillingProductId",       "dataType": "STRING"}
      ]
    }
```

Note

- Pour `"typeName": "AWS:InstanceInformation"`, `InstanceStatus` peut être l'une des options suivantes : Actif, ConnectionLost (connexion perdue), Arrêté, Terminé.
- 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`, 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.
- Certaines métadonnées ne sont pas disponibles à partir des instances Linux. Plus précisément, pour « typeName » : « AWS:Network », les types de métadonnées suivants ne sont pas encore pris en charge pour les instances Linux. Ils SONT pris en charge pour Windows.
  - { "name": "SubnetMask", "dataType": "STRING"},
  - { "name": "DHCPServer", "dataType": "STRING"},
  - { "name": "DNSServer", "dataType": "STRING"},
  - { "name": "Gateway", "dataType": "STRING"},

Utilisation de l'inventaire de fichiers et du registre Windows

AWS Systems Manager Inventory vous permet de rechercher et d'inventorier des fichiers sur les systèmes d'exploitation Windows, Linux et macOS. Vous pouvez également rechercher et inventorier le registre Windows.

Fichiers : Vous pouvez collecter des informations de métadonnées sur les fichiers, notamment leurs noms, leur heure de 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 de fichier des fichiers qui, dans le chemin spécifié, correspondent au modèle. L'inventaire de fichiers utilise l'entrée de paramètre suivante.

```
{
"Path": string,
```

```
"Pattern": array[string],
"Recursive": true,
"DirScanLimit" : number // Optional
}
```

- Chemin : chemin d'accès au répertoire où vous souhaitez 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 des fichiers.
- Récursif : valeur booléenne indiquant si l'inventaire doit parcourir les répertoires de manière récursive.
- DirScanLimit : valeur facultative indiquant le nombre de répertoires à analyser. Utilisez ce paramètre pour minimiser l'impact sur les performances de vos nœuds gérés. 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 et macOS, 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 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.

```
{
"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 les répertoires du registre de manière récursive.
- 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 connexe

AWS Systems ManagerL'inventaire fournit un instantané de votre inventaire actuel pour vous aider à gérer la politique 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 de l'inventaire d'instances gérées Amazon EC2 dans le Manuel du développeur AWS Config.
- AWS Application Discovery Service est conçu pour recueillir l'inventaire sur le type du système d'exploitation, l'inventaire des applications, les processus, les connexions et les métriques de performance du serveur, à partir de vos machines virtuelles sur site et prendre en charge une migration réussie vers AWS. Pour plus d'informations, consultez le Guide de l'utilisateur Application Discovery Service.

Configuration de Systems Manager Inventory

Avant d'utiliser l'inventaire AWS Systems Manager pour la collecte des métadonnées sur les applications, les services, les composants AWS, etc., qui s'exécutent sur vos nœuds gérés, nous vous recommandons de configurer la synchronisation de données de ressources afin de centraliser le stockage de vos données d'inventaire dans un compartiment Amazon Simple Storage Service (Amazon S3) unique. Nous vous recommandons également de configurer la surveillance Amazon EventBridge des événements d'inventaire. Ces processus facilitent l'affichage et la gestion des données d'inventaire et de la collecte.

Rubriques

- Configuration de la synchronisation de données de ressource pour Inventory

- À propos de la surveillance d'événements Inventory par EventBridge

Configuration de la synchronisation de données de ressource pour Inventory

Cette rubrique décrit comment configurer la synchronisation des données de ressource pour l'inventaire AWS Systems Manager . Pour de plus amples informations sur la synchronisation des données de ressource pour Systems Manager Explorer, consultez Configuration de Systems Manager Explorer de sorte à afficher les données de plusieurs comptes et Régions.

À propos de la synchronisation des données de ressource

Vous pouvez utiliser la synchronisation des données de ressources Systems Manager pour envoyer les données d'inventaire collectées à partir de toutes vos nœuds gérés vers un même compartiment Amazon Simple Storage Service (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. Toutes les données d'inventaire étant stockées dans un compartiment Amazon S3 cible, vous pouvez utiliser des services tels qu'Amazon Athena et Amazon QuickSight pour interroger et analyser les données agrégées.

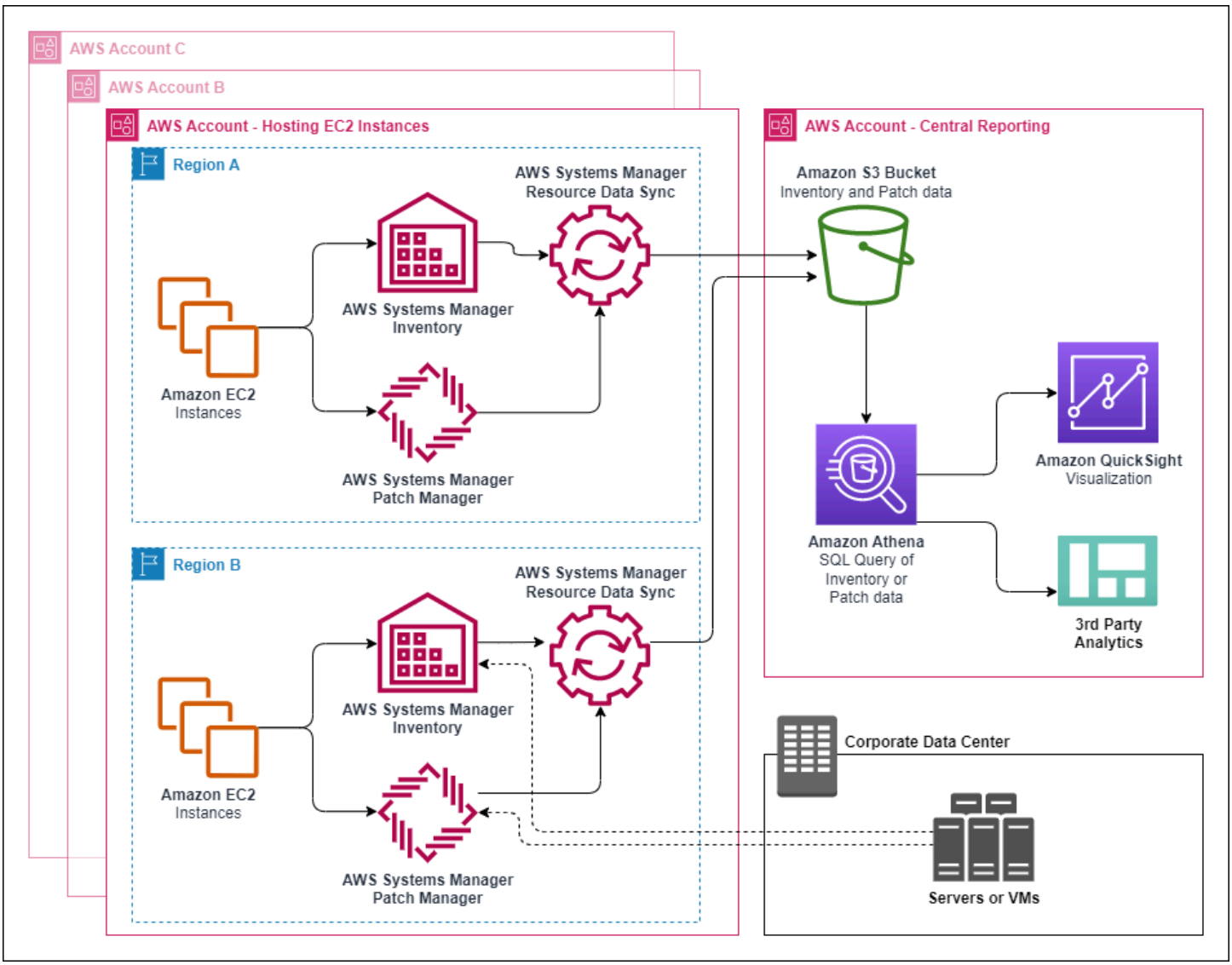
Par exemple, vous avez configure l'inventaire pour la collecte des données relatives au système d'exploitation (OS) et aux applications qui s'exécutent sur une série de 150 nœuds gérés. Certains de ces nœuds sont localisés dans un centre de données sur site, d'autres sont exécutés dans Amazon Elastic Compute Cloud (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 nœud géré, 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 nœuds gérés 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 Amazon S3 cible. Une fois les nouvelles données d'inventaire collectées, Systems Manager met automatiquement à jour les données dans le compartiment Amazon S3. Vous pouvez ensuite transférer les données rapidement et à moindre coût vers Amazon Athena et Amazon. QuickSight

Le diagramme 1 montre comment la synchronisation de données de ressources rassemble les données d'inventaire provenant d'Amazon EC2 et d'autres types de machines dans un

environnement hybride et multicloud vers un compartiment Amazon S3 cible. Ce diagramme montre également comment fonctionne la synchronisation des données de ressources avec plusieurs Comptes AWS et Régions AWS.

Schéma 1 : Synchronisation des données de ressources avec plusieurs Comptes AWS et Régions AWS



Si vous supprimez un nœud géré, la synchronisation des données de ressource conserve le fichier d'inventaire pour le nœud supprimé. Néanmoins, pour les nœuds 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 Amazon S3. Si vous souhaitez suivre l'évolution des stocks au fil du temps, vous pouvez utiliser le AWS Config service pour suivre le type de `SSM:ManagedInstanceInventory` ressource. Pour plus d'informations, consultez Getting Started with AWS Config.

Utilisez les procédures décrites dans cette section pour créer une synchronisation des données de ressources pour Inventory à l'aide d'Amazon S3 et de AWS Systems Manager consoles. Vous pouvez également l'utiliser AWS CloudFormation pour créer ou supprimer une synchronisation des données de ressources. Pour l'utiliser AWS CloudFormation, ajoutez la `AWS::SSM::ResourceDataSync`ressource à votre AWS CloudFormation modèle. Pour de plus amples informations, consultez l'une des ressources de documentation suivantes :

- AWS CloudFormation ressource pour la synchronisation des données des ressources dans AWS Systems Manager (blog)
- Utilisation de modèles AWS CloudFormation dans le Guide de l'utilisateur AWS CloudFormation

> **Note**
>
> Vous pouvez utiliser AWS Key Management Service (AWS KMS) pour chiffrer les données d'inventaire dans le compartiment Amazon S3. Pour un exemple de création d'une synchronisation chiffrée à l'aide de AWS Command Line Interface (AWS CLI) et d'utilisation des données centralisées dans Amazon Athena et Amazon QuickSight, consultez. Démonstration : utiliser la synchronisation de données de ressources pour regrouper les données d'inventaire

Avant de commencer

Avant de créer une synchronisation des données de ressource, appliquez la procédure suivante pour créer un compartiment Amazon S3 central pour stocker les données d'inventaire agrégées. La procédure décrit comment affecter une politique 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 Amazon 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 politique dans la procédure suivante.

> **Note**
>
> Systems Manager Inventory ne peut pas ajouter de données à un compartiment Amazon S3 spécifié si celui-ci est configuré pour utiliser Object Lock. Vérifiez que le compartiment Amazon S3 que vous créez ou sélectionnez pour la synchronisation de données de ressources n'est pas configuré pour utiliser Amazon S3 Object Lock. Pour plus d'informations,

consultez Présentation de la fonctionnalité de verrouillage des objets Amazon S3, consultez le Guide de l'utilisateur d'Amazon Simple Storage Service.

Pour créer et configurer un compartiment Amazon S3 pour la synchronisation de données de ressources

1. Ouvrez la console Amazon S3 sur https://console.aws.amazon.com/s3/.
2. Créez un compartiment pour stocker vos données d'inventaire rassemblées. Pour plus d'informations, consultez Création d'un compartiment dans le Guide de l'utilisateur d'Amazon Simple Storage Service. Notez le nom du bucket et l' Région AWS endroit où vous l'avez créé.
3. Sélectionnez l'onglet Autorisations, puis Politique de compartiment.
4. Copiez et collez la politique de compartiment suivante dans l'éditeur de politique. Remplacez DOC-EXAMPLE-BUCKET et *account-id* par le nom du compartiment S3 que vous avez créé et un identifiant valide. Compte AWS

   Pour permettre Comptes AWS à plusieurs d'envoyer des données d'inventaire au compartiment central Amazon S3, spécifiez chaque compte dans la politique, comme indiqué dans l'`Resource`exemple 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/*"
],
"Condition": {
    "StringEquals": {
         "s3:x-amz-acl": "bucket-owner-full-control",
         "aws:SourceAccount": [
                  "123456789012",
                  "444455556666",
                  "777788889999"
         ]
     },
      "ArnLike": {
          "aws:SourceArn": [
            "arn:aws:ssm:*:123456789012:resource-data-sync/*",
            "arn:aws:ssm:*:444455556666:resource-data-sync/*",
            "arn:aws:ssm:*:777788889999:resource-data-sync/*"
          ]
```

```
    }
}
```

> **Note**
>
> Pour plus d'informations sur la consultation de votre Compte AWS identifiant, consultez l'identifiant de votre compte Amazon Web Services et son alias dans le guide de l'utilisateur IAM.

```
{
    "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/*/accountid=ID_number/*",
            "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*/accountid=ID_number/*",
            "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*/accountid=ID_number/*",
            "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*/accountid=ID_number/*"
        ],
        "Condition": {
            "StringEquals": {
                "s3:x-amz-acl": "bucket-owner-full-control",
                "aws:SourceAccount": "ID_number"
            },
            "ArnLike": {
                "aws:SourceArn": "arn:aws:ssm:*:ID_number:resource-data-sync/*"
```

```
            }
          }
        }
    ]
}
```

Créer une synchronisation de données de ressources pour Inventory

Utilisez la procédure suivante pour créer une synchronisation de données de ressource pour Systems Manager Inventory avec la console Systems Manager. Pour plus d'informations sur la façon de créer une synchronisation des données de ressources à l'aide du AWS CLI, voirDémonstration : Configurer vos nœuds gérés pour l'inventaire à l'aide de l'interface de ligne de commande.

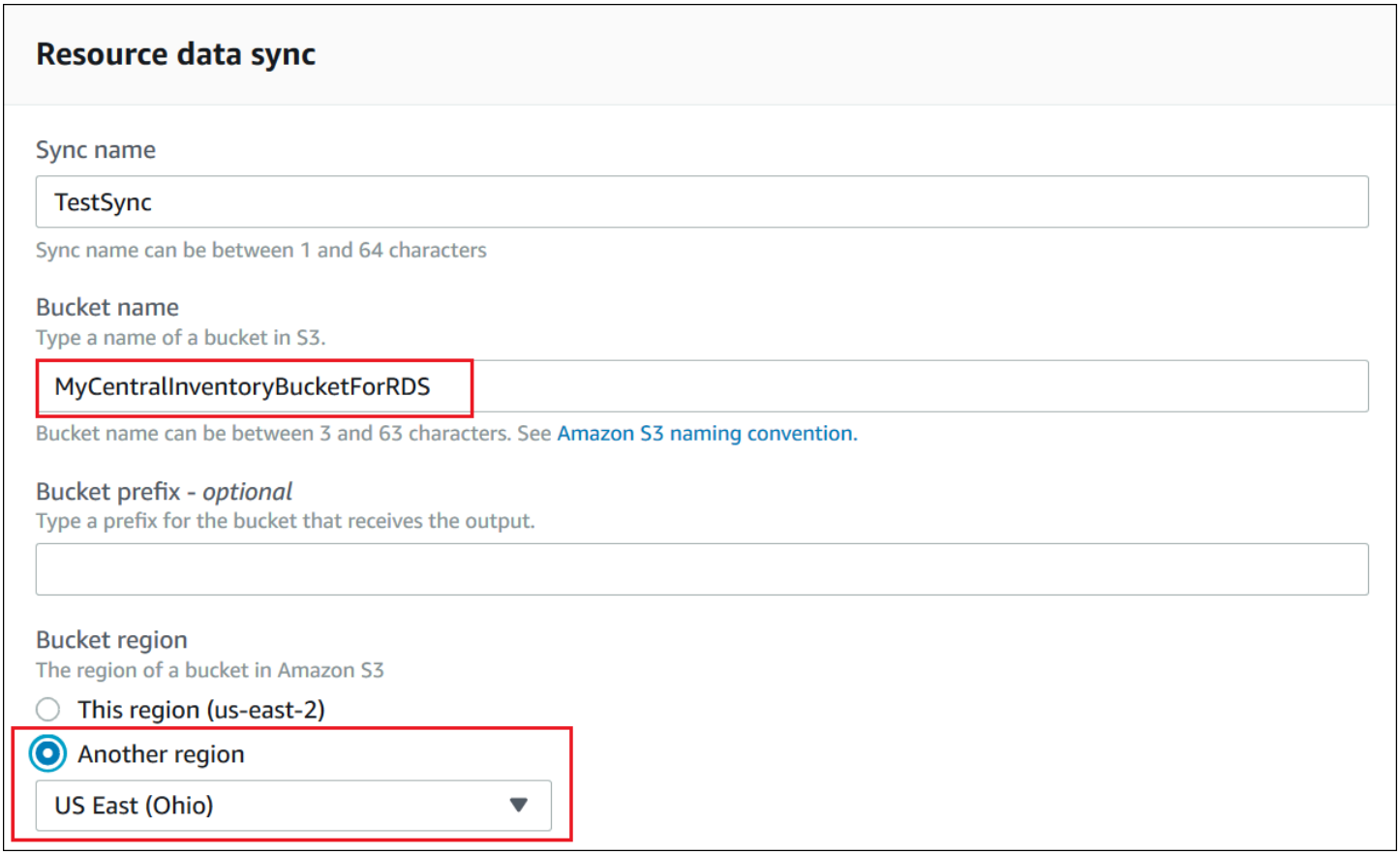
Pour créer une synchronisation de données de ressources

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Dans le menu Account management (Gestion de compte), sélectionnez Resource data sync (Synchronisation de données de ressources).
4. Sélectionnez Create resource data sync (Créer une synchronisation des données de ressource).
5. Dans le champ Sync name (Nom de la synchronisation), saisissez un nom pour la configuration de la synchronisation.
6. Dans le champ Nom du compartiment, saisissez le nom du compartiment Amazon S3 que vous avez créé selon la procédure Pour créer et configurer un compartiment Amazon S3 pour la synchronisation de données de ressources.
7. (Facultatif) Dans le champ Bucket prefix (Préfixe du compartiment), saisissez le nom d'un préfixe de compartiment Amazon S3 (sous-répertoire).
8. Dans le champ Bucket region (Région du compartiment), sélectionnez This region (Cette région) si le compartiment Amazon S3 créé est localisé dans la Région AWS actuelle. Si le compartiment est localisé dans une autre Région AWS, sélectionnez Another region (Autre région), et saisissez le nom de la région.

> **Note**
>
> Si la synchronisation et le compartiment Amazon S3 cible sont localisés dans des régions différentes, vous pourriez être sujet à une tarification de transfert de données. Pour plus d'informations, consultez Tarification Amazon S3.

9. (Facultatif) Dans le champ KMS Key ARN (ARN de clé KMS), saisissez ou collez un ARN de clé KMS pour chiffrer les données d'inventaire dans Amazon S3.
10. Sélectionnez Create (Créer).

Pour synchroniser les données d'inventaire provenant de plusieurs régions Régions AWS, vous devez créer une synchronisation des données de ressources dans chaque région. Répétez cette procédure dans chaque Région AWS endroit où vous souhaitez collecter des données d'inventaire et les envoyer au compartiment central Amazon S3. Lorsque vous créez la synchronisation dans chaque région, spécifiez le compartiment Amazon S3 central dans le champ Bucket name (Nom du compartiment). Ensuite, utilisez l'option Bucket region (Région du 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 Amazon S3 central.



Création d'une synchronisation des données de ressource d'inventaire pour les comptes définis dans AWS Organizations

Vous pouvez synchroniser les données d'inventaire Comptes AWS définies dans AWS Organizations un compartiment Amazon S3 central. Après avoir terminé les procédures suivantes, les données d'inventaire sont synchronisées avec des préfixes de clé Amazon S3 individuels dans le compartiment central. Chaque préfixe de clé représente un identifiant différent. Compte AWS

Avant de commencer

Avant de commencer, vérifiez que vous avez configuré et configuré Comptes AWS dans AWS Organizations. Pour plus d'informations, consultez dans le Guide de l'utilisateur AWS Organizations .

Sachez également que vous devez créer la synchronisation des données des ressources basée sur l'organisation pour chacune d'elles Région AWS et Compte AWS définie dans. AWS Organizations

Création d'un compartiment Amazon S3 central

Appliquez la procédure suivante pour créer un compartiment Amazon S3 central pour stocker les données d'inventaire agrégées. La procédure décrit comment affecter une politique de compartiment

qui permet à Systems Manager d'écrire des données d'inventaire dans le compartiment à partir de votre ID de compte AWS Organizations . Si vous disposez déjà d'un compartiment Amazon 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 politique dans la procédure suivante.

Pour créer et configurer un compartiment Amazon S3 pour la synchronisation des données de ressources pour plusieurs comptes définis dans AWS Organizations

1. Ouvrez la console Amazon S3 sur https://console.aws.amazon.com/s3/.
2. Créez un compartiment pour stocker vos données d'inventaire agrégées. Pour plus d'informations, consultez Création d'un compartiment dans le Guide de l'utilisateur d'Amazon Simple Storage Service. Notez le nom du bucket et l' Région AWS endroit où vous l'avez créé.
3. Sélectionnez l'onglet Autorisations, puis Politique de compartiment.
4. Copiez et collez la politique de compartiment suivante dans l'éditeur de politique. Remplacez DOC-EXAMPLE-BUCKET et *organization-id par* le nom du compartiment Amazon S3 que vous avez créé et un ID de compte valide. AWS Organizations

   Vous avez également la possibilité de remplacer *bucket-prefix* par le nom d'un préfixe Amazon S3 (sous-répertoire). Si vous n'avez pas créé de préfixe, supprimez *bucket-prefix/* de l'ARN dans la politique suivante.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "SSMBucketPermissionsCheck",
      "Effect": "Allow",
      "Principal": {
        "Service": "ssm.amazonaws.com"
      },
      "Action": "s3:GetBucketAcl",
      "Resource": "arn:aws:s3:::S3_bucket_name"
    },
    {
      "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",
          "aws:SourceOrgID": "organization-id"
                  }
      }
    },
    {
      "Sid": " SSMBucketDeliveryTagging",
      "Effect": "Allow",
      "Principal": {
        "Service": "ssm.amazonaws.com"
      },
      "Action": "s3:PutObjectTagging",
      "Resource": [
        "arn:aws:s3:::DOC-EXAMPLE-BUCKET/bucket-prefix/*/accountid=*/*"
      ]
    }
  ]
}
```

Créer une synchronisation des données de ressource d'inventaire pour les comptes définis dans AWS Organizations

La procédure suivante décrit comment utiliser le AWS CLI pour créer une synchronisation des données de ressources pour les comptes définis dans AWS Organizations. Vous devez utiliser le AWS CLI pour effectuer cette tâche. Vous devez également exécuter cette procédure pour chacun d'entre eux Région AWS et Compte AWS définie dans AWS Organizations.

Pour créer une synchronisation des données de ressource pour les comptes définis dans AWS Organizations (AWS CLI)

1. Installez et configurez le AWS Command Line Interface (AWS CLI), si ce n'est pas déjà fait.

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l' AWS CLI.

2. Exécutez la commande suivante pour vérifier qu'aucune autre synchronisation de données de ressources n'est effectuée. Vous ne pouvez avoir qu'une seule synchronisation de données de ressources basée sur l'organisation.

   ```
   aws ssm list-resource-data-sync
   ```

   Si la commande renvoie une autre synchronisation de données de ressources, vous devez la supprimer ou choisir de ne pas en créer de nouvelle.

3. Exécutez la commande suivante pour créer une synchronisation des données de ressource pour un compte défini dans AWS Organizations. Pour DOC-EXAMPLE-BUCKET, spécifiez le nom du compartiment Amazon S3 que vous avez précédemment créé dans cette rubrique. Si vous avez créé un préfixe (sous-répertoire) pour votre compartiment, indiquez ces informations dans *prefix-name*.

   ```
   aws ssm create-resource-data-sync --sync-name name --s3-
   destination "BucketName=DOC-EXAMPLE-BUCKET,Prefix=prefix-
   name,SyncFormat=JsonSerDe,Region=Région AWS, for example us-
   east-2,DestinationDataSharing={DestinationDataSharingType=Organization}"
   ```

4. Répétez les étapes 2 et 3 pour chaque Région AWS Compte AWS endroit où vous souhaitez synchroniser les données avec le compartiment Amazon S3 central.

Gestion des synchronisations des données de ressource

Chacun Compte AWS peut avoir 5 synchronisations de données de ressources par personne. Région AWS Vous pouvez utiliser la console AWS Systems ManagerFleet Manager pour gérer les synchronisations des données de vos ressources.

Pour afficher les synchronisations des données de ressource

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Dans le menu déroulant Gestion de compte, sélectionnez Synchronisation de données de ressource.

4. Sélectionnez une synchronisation des données de ressource dans le tableau, puis choisissez Afficher les détails pour afficher les informations relatives à la synchronisation de vos données de ressource.

Pour supprimer une synchronisation de données de ressources

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Dans le menu déroulant Gestion de compte, sélectionnez Synchronisation de données de ressource.
4. Sélectionnez une synchronisation des données de ressource dans le tableau, puis choisissez Supprimer.

À propos de la surveillance d'événements Inventory par EventBridge

Vous pouvez configurer une règle dans Amazon EventBridge pour créer un événement en réponse à des changements d'état de ressources AWS Systems Manager Inventory. EventBridge prend en charge des événements pour les changements d'état suivants de ressources Inventory. Tous les événements sont générés sur la base du meilleur effort.

Type d'inventaire personnalisé supprimé pour une instance spécifique : si une règle est configurée pour surveiller cet événement, EventBridge crée un événement lorsqu'un type d'inventaire personnalisé sur un nœud spécifique est supprimé. EventBridge envoie un événement par nœud pour chaque type d'inventaire personnalisé. Voici un exemple de modèle d'événement.

```
{
    "timestampMillis": 1610042981103,
    "source": "SSM",
    "account": "123456789012",
    "type": "INVENTORY_RESOURCE_STATE_CHANGE",
    "startTime": "Jan 7, 2021 6:09:41 PM",
    "resources": [
        {
            "arn": "arn:aws:ssm:us-east-1:123456789012:managed-instance/i-12345678"
        }
    ],
    "body": {
```

```
        "action-status": "succeeded",
        "action": "delete",
        "resource-type": "managed-instance",
        "resource-id": "i-12345678",
        "action-reason": "",
        "type-name": "Custom:MyCustomInventoryType"
    }
}
```

Type d'inventaire personnalisé supprimé pour toutes les instances : si une règle est configurée pour surveiller cet événement, EventBridge crée un événement lorsqu'un type d'inventaire personnalisé sur tous les nœuds est supprimé. Voici un exemple de modèle d'événement.

```
{
    "timestampMillis": 1610042904712,
    "source": "SSM",
    "account": "123456789012",
    "type": "INVENTORY_RESOURCE_STATE_CHANGE",
    "startTime": "Jan 7, 2021 6:08:24 PM",
    "resources": [

    ],
    "body": {
        "action-status": "succeeded",
        "action": "delete-summary",
        "resource-type": "managed-instance",
        "resource-id": "",
        "action-reason": "The delete for type name Custom:SomeCustomInventoryType
 was completed. The deletion summary is: {\"totalCount\":1,\"remainingCount\":0,
\"summaryItems\":[{\"version\":\"1.1\",\"count\":1,\"remainingCount\":0}]}",
        "type-name": "Custom:MyCustomInventoryType"
    }
}
```

Appel PutInventory avec un événement de l'ancienne version de schéma : si une règle est configurée pour surveiller cet événement, EventBridge crée un événement lors d'un appel PutInventory utilisant une version de schéma inférieure au schéma actuel. Cet événement s'applique à tous les types d'inventaire. Voici un exemple de modèle d'événement.

```
{
    "timestampMillis": 1610042629548,
    "source": "SSM",
```

```
    "account": "123456789012",
    "type": "INVENTORY_RESOURCE_STATE_CHANGE",
    "startTime": "Jan 7, 2021 6:03:49 PM",
    "resources": [
        {
            "arn": "arn:aws:ssm:us-east-1:123456789012:managed-instance/i-12345678"
        }
    ],
    "body": {
        "action-status": "failed",
        "action": "put",
        "resource-type": "managed-instance",
        "resource-id": "i-01f017c1b2efbe2bc",
        "action-reason": "The inventory item with type name
 Custom:MyCustomInventoryType was sent with a disabled schema verison 1.0. You must
 send a version greater than 1.0",
        "type-name": "Custom:MyCustomInventoryType"
    }
}
```

Pour obtenir des informations sur la façon de configurer EventBridge afin de surveiller ces événements, veuillez consulter Configurer EventBridge pour des événements Systems Manager.

Configuration de la collecte d'inventaire

Cette section décrit comment configurer la collecte AWS Systems Manager d'inventaire sur un ou plusieurs nœuds gérés à l'aide de la console Systems Manager. Pour un exemple de configuration de la collecte d'inventaire à l'aide de AWS Command Line Interface (AWS CLI), consultezProcédures Systems Manager Invetory.

Lorsque vous configurez la collecte d'inventaire, vous commencez par créer une AWS Systems Manager State Manager association. Systems Manager collecte les données d'inventaire lorsque l'association est exécutée. Si vous ne créez pas d'abord l'association et que vous tentez d'appeler le `aws:softwareInventory` plugin en utilisant, par exemple AWS Systems Manager Run Command, le système renvoie l'erreur suivante : `The aws:softwareInventory plugin can only be invoked via ssm-associate.`

> **Note**
>
> Vous devez connaître le comportement suivant en cas de création d'associations d'inventaire multiples pour un nœud géré.

- Chaque nœud peut se voir attribuer une association d'inventaire qui cible tous les nœuds (--targets « Key=InstanceIds, Values=* »).
- Chaque nœud peut également se voir attribuer une association spécifique qui utilise soit des paires clé/valeur de balise, soit un groupe de AWS ressources.
- Si plusieurs associations d'inventaire sont affectées à un nœud, l'association qui ne s'est pas exécutée présente le statut Skipped (Ignoré). L'association qui s'est exécutée la plus récemment affiche le statut réel de l'association d'inventaire.
- Si un nœud se voit affecter plusieurs associations d'inventaire et que chacune utilise une paire clé-valeur de balise, le conflit de balises empêche ces associations d'inventaire de s'exécuter sur le nœud. L'association s'exécute normalement sur les nœuds exempts du conflit clé-valeur de balise.

Avant de commencer

Avant de configurer la collecte d'inventaire, effectuez les tâches suivantes.

- Mettez à AWS Systems Manager SSM Agent jour les nœuds que vous souhaitez inventorier. En exécutant la dernière version de SSM Agent, 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'SSM Agent à l'aide de State Manager, consultez Démonstration : Mise à jour automatique de l'SSM Agent (CLI).
- Vérifiez que vous avez satisfait la configuration requise pour vos instances Amazon Elastic Compute Cloud (Amazon EC2) et vos machines non EC2 dans un environnement hybride et multicloud. Pour plus d'informations, veuillez consulter Con AWS Systems Manager figuration.
- Pour les nœuds Microsoft Windows, vérifiez que votre nœud géré est configuré avec Windows PowerShell 3.0 (ou version ultérieure). SSM Agentutilise l'`ConvertTo-Json`applet de commande PowerShell pour convertir les données d'inventaire Windows Update au format requis.
- (Facultatif) Créez une synchronisation de données de ressources pour stocker les données d'inventaire de façon centralisée dans un compartiment Amazon S3. La synchronisation de données de ressources met alors automatiquement à jour les données centralisées lorsque de nouvelles données d'inventaire sont collectées. Pour plus d'informations, consultez Configuration de la synchronisation de données de ressource pour Inventory.
- (Facultatif) Créez un fichier JSON pour collecter l'inventaire personnalisé. Pour plus d'informations, consultez Utilisation de l'inventaire personnalisé.

Répertoriez tous les nœuds gérés de votre Compte AWS

Vous pouvez inventorier tous les nœuds gérés de votre Compte AWS site en créant une association d'inventaire globale. Une association d'inventaire global effectue les actions suivantes :

- Applique automatiquement la configuration de l'inventaire global (association) à tous les nœuds gérés existants de votre Compte AWS. Les nœuds gérés disposant déjà d'une association d'inventaire sont ignorés lorsque l'association d'inventaire global est appliquée et s'exécute. Lorsqu'un nœud est ignoré, le message de statut détaillé indique `Overridden By Explicit Inventory Association`. Ces nœuds sont ignorés par l'association globale, mais ils continuent de générer des inventaires lorsqu'ils exécutent l'association d'inventaire qui leur est affectée.
- Ajoute automatiquement les nouveaux nœuds créés dans votre Compte AWS entreprise à l'association d'inventaire globale.

> **Note**
>
> - Si un nœud géré est configuré pour l'association d'inventaire global et que vous lui affectez une association spécifique, alors Systems Manager Inventory annulera la priorité de l'association globale et appliquera l'association spécifique.
> - Les associations d'inventaire global sont disponibles dans SSM Agent, version 2.0 790.0 ou version ultérieure. Pour plus d'informations sur la mise à jour de l'SSM Agent sur vos nœuds, consultez Mise à jour de SSM Agent à l'aide de Run Command.

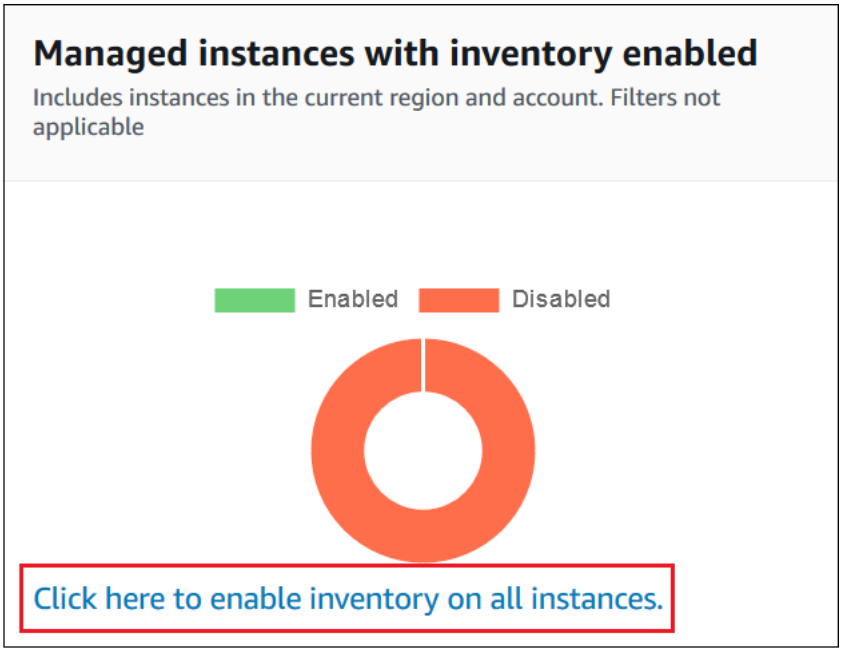
Configuration de la collecte d'inventaire en un clic (console)

Utilisez la procédure suivante pour configurer Systems Manager Inventory pour tous les nœuds gérés de votre Compte AWS et en un seul Région AWS.

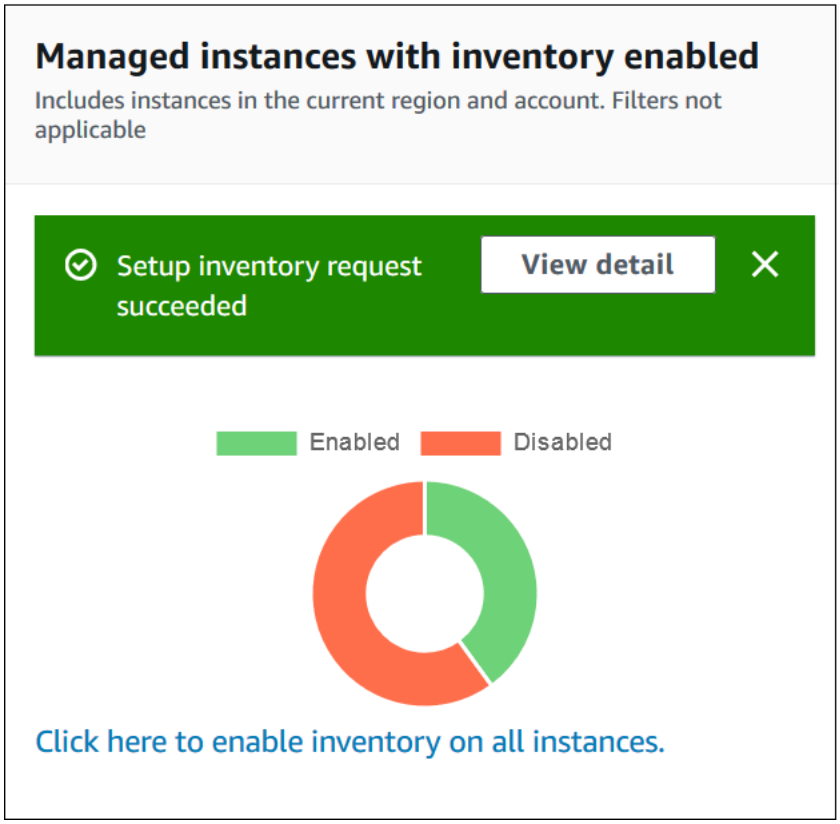
Pour configurer l'ensemble de vos nœuds gérés dans la région actuelle pour Systems Manager Inventory

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le volet de navigation, sélectionnez Inventory.

3. Dans la carte Managed instances with inventory enabled (Instances gérées avec l'inventaire activé), sélectionnez 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 de nœuds gérés 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 nœuds gérés.

Configuration de la collecte à l'aide de la console

Cette section inclut des informations sur la configuration de Systems Manager Inventory pour qu'il collecte les métadonnées de vos nœuds gérés à l'aide de la console Systems Manager. Vous pouvez collecter rapidement des métadonnées à partir de tous les nœuds d'un compte spécifique Compte AWS (et de tous les futurs nœuds qui pourraient être créés dans ce compte) ou vous pouvez collecter des données d'inventaire de manière sélective à l'aide de balises ou d'identifiants de nœuds.

> **Note**
>
> Avant de terminer cette procédure, vérifiez si une association d'inventaire global existe déjà. Si une association d'inventaire global existe déjà, chaque fois que vous lancez une nouvelle instance, l'association lui sera appliquée et la nouvelle instance sera inventoriée.

Pour configurer la collecte d'inventaire

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le volet de navigation, sélectionnez Inventory.
3. Sélectionnez Setup Inventory (Configurer l'inventaire).
4. Dans la section Targets (Cibles), identifiez les nœuds sur lesquels 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 tous les nœuds gérés ne disposant d'aucune association d'inventaire existante. Si vous sélectionnez cette option, les nœuds disposant déjà d'associations d'inventaire sont ignorés lors de la collecte d'inventaire, et s'affichent avec un statut Skipped (Ignoré) dans les résultats d'inventaire. Pour plus d'informations, consultez Répertoriez tous les nœuds gérés de votre Compte AWS.

- Specifying a tag (Spécification d'une balise) : utilisez cette option afin de spécifier une balise unique pour identifier les nœuds de votre compte à partir desquels vous souhaitez collecter l'inventaire. Si vous utilisez une balise, tous les nœuds créés ensuite avec la même balise généreront également des rapports d'inventaires. S'il existe déjà une association d'inventaire avec tous les nœuds, l'utilisation d'une balise pour sélectionner des nœuds spécifiques en tant que cible pour un autre inventaire remplace l'appartenance du nœud dans le groupe cible All managed instances (Toutes les instances gérées). Les nœuds gérés 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) : utilisez cette option pour choisir des nœuds gérés 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). Le nœud est ignoré lors des futures collectes d'inventaire à partir de All managed instances (Toutes les instances gérées).

> **Note**
>
> Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés pour obtenir des conseils de dépannage.

5. Dans la section Schedule (Calendrier), sélectionnez la fréquence à laquelle le système doit collecter les métadonnées d'inventaire à partir de vos nœuds.
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.

   Dépôt de

   - Sous Linux et macOS, 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.

7. Dans la section Advanced (Avancé), sélectionnez Sync inventory execution logs to an Amazon S3 bucket (Synchroniser les journaux d'exécution d'inventaire de synchronisation sur un compartiment Amazon S3) si vous souhaitez stocker le statut d'exécution d'association dans un compartiment Amazon S3.
8. Sélectionnez Setup Inventory (Configurer l'inventaire). Systems Manager crée une association State Manager et exécute immédiatement l'inventaire sur les nœuds.

9. Dans le panneau de navigation, sélectionnez 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 Amazon S3 bucket (Synchroniser les journaux d'exécution d'inventaire sur un compartiment Amazon S3), vous pouvez afficher les données des journaux dans Amazon S3 après quelques minutes. Si vous souhaitez afficher les données d'inventaire pour un nœud spécifique, sélectionnez Managed Instances (Instances gérées) dans le panneau de navigation.
10. Sélectionnez Manage (Gérer), puis View details (Afficher les détails).
11. Sur la page des détails du nœud, sélectionnez 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 Inventory AWS Systems Manager.

Rubriques

- Interrogation des données d'inventaire à partir de plusieurs régions et comptes
- Interrogation d'une collecte d'inventaire à l'aide de filtres
- Agrégation des données d'inventaire

Interrogation des données d'inventaire à partir de plusieurs régions et comptes

AWS Systems Manager Inventory s'intègre à Amazon Athena pour vous aider à interroger les données d'inventaire provenant de plusieurs Régions AWS et. Comptes AWS L'intégration d'Athena utilise la synchronisation des données des ressources afin que vous puissiez consulter les données d'inventaire de tous vos nœuds gérés sur la page d'affichage détaillé de la AWS Systems Manager console.

> **Important**
>
> Cette fonctionnalité permet AWS Glue d'explorer les données de votre compartiment Amazon Simple Storage Service (Amazon S3), et d'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 crawlers (découverte de données) et les tâches ETL (traitement et chargement de 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 Systems Manager Inventory. Pour en savoir plus, consultez la tarification Amazon Athena et la tarification AWS Glue.

Vous pouvez afficher les données d'inventaire sur la page Detailed View (Vue détaillée) dans toutes les Régions AWS où Amazon Athena est disponible. Pour obtenir une liste des régions prises en charge, veuillez consulter la rubrique Points de terminaison de service Amazon Athena de la 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 Configuration de la synchronisation de données de ressource pour Inventory.

De plus, sachez que la page Detailed View (Vue détaillée) affiche les données d'inventaire pour le propriétaire du compartiment Amazon S3 central utilisé par la synchronisation de données de ressources. Si vous n'êtes pas le propriétaire du compartiment Amazon S3 central, vous ne verrez pas les données d'inventaire sur la page Detailed View (Vue détaillée).

Configuration de l'accès

Avant de pouvoir interroger et afficher des données de plusieurs comptes et régions sur la page Vue détaillée dans la console Systems Manager, vous devez configurer votre entité IAM avec l'autorisation d'afficher les données.

Si les données d'inventaire sont stockées dans un compartiment Amazon S3 qui utilise le chiffrement AWS Key Management Service (AWS KMS), vous devez également configurer votre entité IAM et le rôle de `Amazon-GlueServiceRoleForSSM` service pour le AWS KMS chiffrement.

Rubriques

- Configuration de votre entité IAM pour accéder à la page Vue détaillée
- (Facultatif) Configurer les autorisations d'affichage des données AWS KMS chiffrées

Configuration de votre entité IAM pour accéder à la page Vue détaillée

Ce qui suit décrit les autorisations minimales requises pour afficher les données d'inventaire sur la page Affichage détaillé.

La politique gérée **`AWSQuicksightAthenaAccess`**

Le `PassRole` suivant et les blocs d'autorisations requis supplémentaires

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowGlue",
            "Effect": "Allow",
            "Action": [
                "glue:GetCrawler",
                "glue:GetCrawlers",
                "glue:GetTables",
                "glue:StartCrawler",
                "glue:CreateCrawler"
            ],
            "Resource": "*"
        },
        {
            "Sid": "iamPassRole",
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": "glue.amazonaws.com"
                }
            }
        },
        {
            "Sid": "iamRoleCreation",
           "Effect": "Allow",
            "Action": [
                "iam:CreateRole",
                "iam:AttachRolePolicy"
            ],
            "Resource": "arn:aws:iam::account_ID:role/*"
        },
```

```
        {
            "Sid": "iamPolicyCreation",
            "Effect": "Allow",
            "Action": "iam:CreatePolicy",
            "Resource": "arn:aws:iam::account_ID:policy/*"
        }
    ]
}
```

(Facultatif) Si le compartiment Amazon S3 utilisé pour stocker les données d'inventaire est chiffré à l'aide de cette méthode AWS KMS, vous devez également ajouter le bloc suivant à la politique.

```
{
    "Effect": "Allow",
    "Action": [
        "kms:Decrypt"
    ],
    "Resource": [
        "arn:aws:kms:Region:account_ID:key/key_ARN"
    ]
}
```

Pour activer l'accès, ajoutez des autorisations à vos utilisateurs, groupes ou rôles :

- Utilisateurs et groupes dans AWS IAM Identity Center :

  Créez un jeu d'autorisations. Suivez les instructions de la rubrique Création d'un jeu d'autorisations du Guide de l'utilisateur AWS IAM Identity Center .

- Utilisateurs gérés dans IAM par un fournisseur d'identité :

  Créez un rôle pour la fédération d'identité. Pour plus d'informations, voir la rubrique Création d'un rôle pour un fournisseur d'identité tiers (fédération) du Guide de l'utilisateur IAM.

- Utilisateurs IAM :
  - Créez un rôle que votre utilisateur peut assumer. Suivez les instructions de la rubrique Création d'un rôle pour un utilisateur IAM du Guide de l'utilisateur IAM.
  - (Non recommandé) Attachez une politique directement à un utilisateur ou ajoutez un utilisateur à un groupe d'utilisateurs. Suivez les instructions de la rubrique Ajout d'autorisations à un utilisateur (console) du Guide de l'utilisateur IAM.

(Facultatif) Configurer les autorisations d'affichage des données AWS KMS chiffrées

Si le compartiment Amazon S3 utilisé pour stocker les données d'inventaire est chiffré à l'aide du AWS Key Management Service (AWS KMS), vous devez configurer votre entité IAM et le rôle GlueServiceRoleForAmazon-SSM avec `kms:Decrypt` des autorisations pour la AWS KMS clé.

Avant de commencer

Pour fournir les `kms:Decrypt` autorisations relatives à la AWS KMS clé, ajoutez le bloc de politique suivant à votre entité IAM :

```
{
    "Effect": "Allow",
    "Action": [
        "kms:Decrypt"
    ],
    "Resource": [
        "arn:aws:kms:Region:account_ID:key/key_ARN"
    ]
}
```

Si ce n'est pas déjà fait, suivez cette procédure et ajoutez `kms:Decrypt` des autorisations pour la AWS KMS clé.

Utilisez la procédure suivante pour configurer le rôle GlueServiceRoleForAmazon-SSM avec des `kms:Decrypt` autorisations pour la AWS KMS clé.

Pour configurer le rôle GlueServiceRoleForAmazon-SSM avec des autorisations **`kms:Decrypt`**

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le volet de navigation, choisissez Roles, puis utilisez le champ de recherche pour localiser le rôle GlueServiceRoleForAmazon-SSM. La page Récapitulatif s'ouvre.
3. Utilisez le champ de recherche pour trouver le rôle GlueServiceRoleForAmazon-SSM. Sélectionnez le nom de rôle. La page Récapitulatif s'ouvre.
4. Sélectionnez le nom de rôle. La page Récapitulatif s'ouvre.
5. Sélectionnez Ajouter une politique en ligne. La page Créer une politique s'ouvre.
6. Sélectionnez l'onglet JSON.
7. Supprimez le texte JSON existant dans l'éditeur, puis copiez et collez la politique suivante dans l'éditeur JSON.

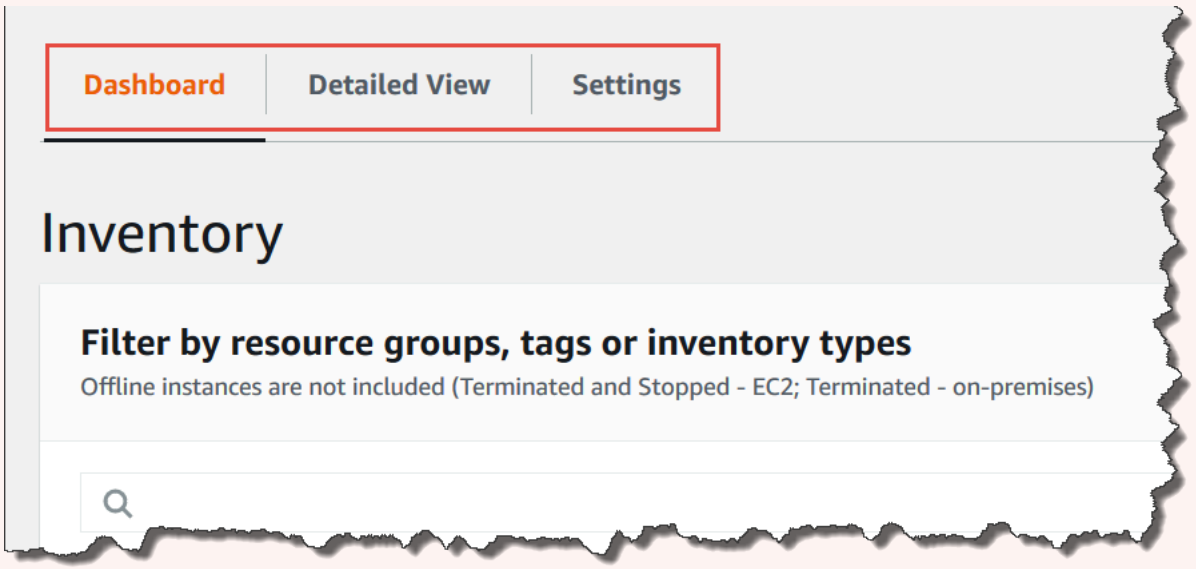
```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": [
                "arn:aws:kms:Region:account_ID:key/key_ARN"
            ]
        }
    ]
}
```

8. Choisissez Review policy (Examiner une politique)
9. Sur la page Examiner une politique, entrez un nom dans le champ Nom.
10. Sélectionnez Créer une politique.

Interrogation des données sur la page Vue détaillée d'inventaire

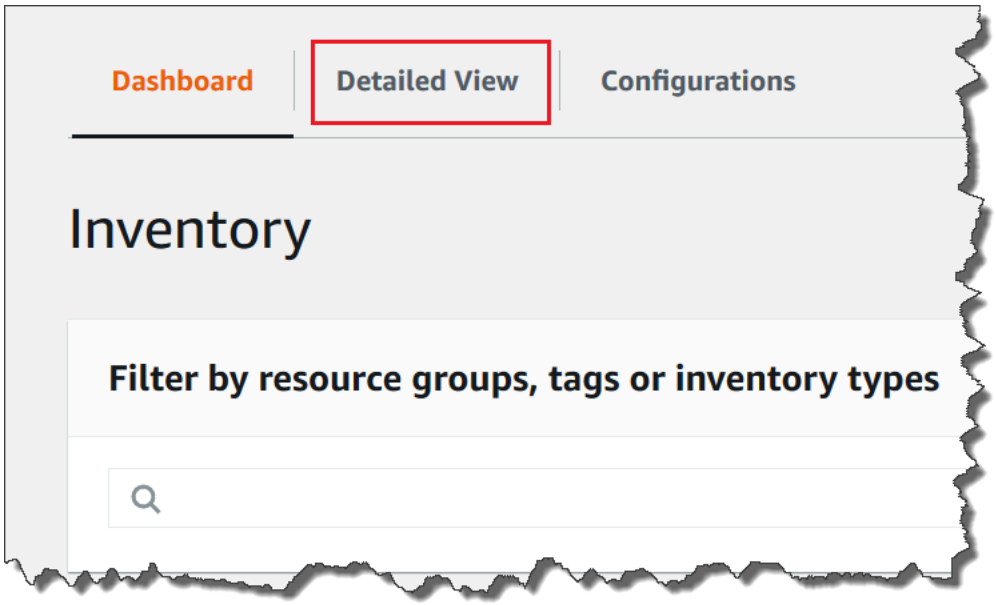
Utilisez la procédure suivante pour afficher les données d'inventaire provenant de plusieurs sources Régions AWS et Comptes AWS sur la page de vue détaillée de l'inventaire de Systems Manager.

> **Important**
>
> La page Detailed View (Vue détaillée) est uniquement disponible dans les Régions AWS qui proposent Amazon Athena. Si les onglets suivants ne sont pas affichés sur la page Systems Manager Inventory, 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.

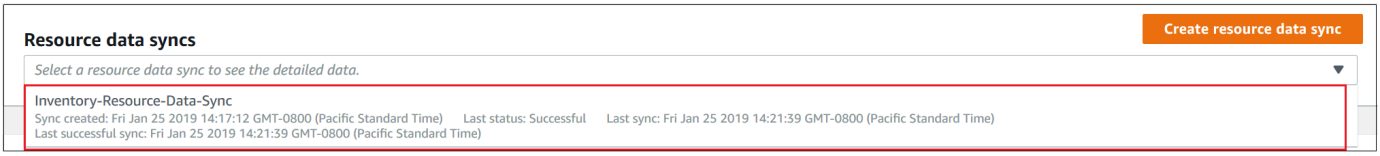


Pour afficher les données d'inventaire à partir de plusieurs régions et comptes dans la console AWS Systems Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le volet de navigation, sélectionnez Inventory.
3. Sélectionnez l'onglet Detailed View (Vue détaillée).

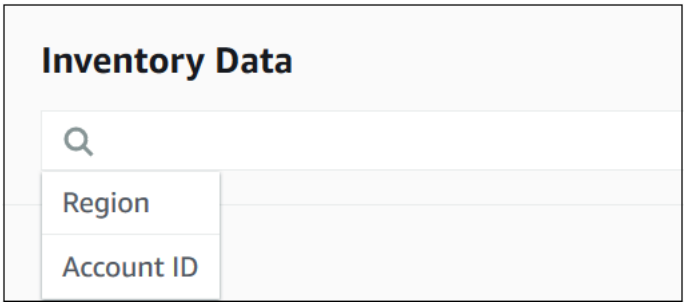


4. Sélectionnez la synchronisation de données de ressources pour laquelle vous souhaitez rechercher des données.



5. Dans la liste Inventory Type (Type d'inventaire), sélectionnez le type de données d'inventaire que vous souhaitez interroger et appuyez sur Enter.

6. Pour filtrer les données, sélectionnez la barre de filtre, puis sélectionnez une option de filtre.



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 du crawler AWS Glue

AWS Glue analyse les données d'inventaire dans le compartiment central Amazon S3 deux fois par jour, par défaut. Si vous modifiez fréquemment les types de données à collecter sur vos nœuds, vous préférerez peut-être analyser les données plus fréquemment, comme cela est décrit dans la procédure suivante.

> **Important**
>
> AWS Glue vous facture sur la Compte AWS base d'un taux horaire, facturé à la seconde, pour les robots d'exploration (découverte de données) et les tâches ETL (traitement et

chargement de données). Avant de modifier la planification du crawler, consultez la page Tarification AWS Glue.

Pour modifier la planification du crawler de données d'inventaire

1. Ouvrez la AWS Glue console à l'adresse https://console.aws.amazon.com/glue/.
2. Dans le panneau de navigation, sélectionnez Crawlers. (Analyseurs)
3. Dans la liste des crawlers, sélectionnez l'option à côté du crawler de données Systems Manager Inventory. Le nom du crawler utilise le format suivant :

   `AWSSystemsManager-DOC-EXAMPLE-BUCKET-`*`Region`*`-`*`account_ID`*
4. Sélectionnez Action, puis Modifier un crawler.
5. Dans le panneau de navigation, sélectionnez 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 crawlers dans le Guide du développeur AWS Glue .

> **Important**
>
> Vous pouvez suspendre le robot d'exploration pour ne plus être débité. AWS Glue Si vous suspendez le crawler 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 les données d'inventaire collectées, vous pouvez utiliser les fonctionnalités de filtre dans AWS Systems Manager pour interroger une liste de nœuds gérés respectant certains critères de filtre.

Pour interroger des nœuds d'après des filtres d'inventaire

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le volet de navigation, sélectionnez Inventory.

3. Dans la section Filter by resource groups, tags or inventory types (Filtrer par groupes de ressources, balises ou types d'inventaire), sélectionnez la zone de filtre. Une liste de filtres prédéfinis s'affiche.
4. Sélectionnez un attribut sur lequel filtrer. Par exemple, sélectionnez **`AWS:Application`**. Si vous y êtes invité, sélectionnez un attribut secondaire sur lequel filtrer. Par exemple, sélectionnez **`AWS:Application.Name`**.
5. Sélectionnez un délimiteur dans la liste. Par exemple, sélectionnez 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'SSM Agent est nommé Amazon SSM Agent).
7. Appuyez sur Enter. Le système renvoie une liste de nœuds gérés comprenant un nom d'application qui commence par le terme Amazon.

> **Note**
>
> Vous pouvez combiner plusieurs filtres pour affiner votre recherche.

Agrégation des données d'inventaire

Une fois vos nœuds gérés configurés 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 de nœuds gérés pour collecter le type d'inventaire `AWS:Application`. En utilisant les informations de cette section, vous pouvez voir le décompte exact des nœuds configurés 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 de nœuds qui exécutent Windows, le nombre de nœuds qui exécutent Linux et le nombre de nœuds qui exécutent macOS.

Les procédures de cette section expliquent comment afficher les décomptes agrégés de données d'inventaire à l'aide de l'AWS Command Line Interface (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 :

- Si vous résiliez un nœud géré configuré pour collecter des données d'inventaire, Systems Manager conserve les données d'inventaire pendant 30 jours, puis les supprime. Pour les nœuds en cours d'exécution, le système supprime les données d'inventaire antérieures à 30 jours. Si vous devez stocker les données d'inventaire plus de 30 jours, vous pouvez utiliser AWS Config pour enregistrer l'historique ou pour interroger régulièrement les données et les charger dans un compartiment Amazon Simple Storage Service (Amazon S3).
- Si un nœud a été préalablement configuré 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 le nœud soit mis hors service et une fois le délai des 30 jours passé.

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 tous les nœuds futurs susceptibles d'être créés dans ce compte), consultez Configuration de la collecte à l'aide de la console.

Rubriques

- Agrégation des données d'inventaire pour afficher les décomptes de nœuds qui collectent des types spécifiques de données
- Agrégation des données d'inventaire avec des groupes pour voir quels nœuds sont ou non configurés pour collecter un type d'inventaire

Agrégation des données d'inventaire pour afficher les décomptes de nœuds qui collectent des types spécifiques de données

Vous pouvez utiliser l'opération d'API AWS Systems Manager GetInventory pour afficher les décomptes agrégés de nœuds qui collectent un ou plusieurs types d'inventaire et types de données. Par exemple, le type d'inventaire `AWS:InstanceInformation` vous permet d'afficher une vue agrégée des systèmes d'exploitation en utilisant l'opération API GetInventory 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 retourne des informations telles que les suivantes.

```
{
   "Entities":[
      {
         "Data":{
            "AWS:InstanceInformation":{
               "Content":[
                  {
                     "Count":"7",
                     "PlatformType":"windows"
                  },
                  {
                     "Count":"5",
                     "PlatformType":"linux"
                  }
               ]
            }
         }
      }
   ]
}
```

Démarrer

Déterminez les types d'inventaire et les types de données pour lesquels vous souhaitez afficher des décomptes. Vous pouvez consulter la liste des types d'inventaires 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 champ TypeName montre 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.

Exemple 1

Cet exemple agrège un décompte des rôles Windows utilisés par vos nœuds.

```
aws ssm get-inventory --aggregators "Expression=AWS:WindowsRole.Name"
```

Exemple 2

Cet exemple agrège un décompte des applications installées sur vos nœuds.

```
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.

Exemple 1

Cet exemple agrège un décompte des types de système d'exploitation utilisés par vos nœuds. 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 nœuds 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 telles que les 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 quels nœuds sont ou non configurés pour collecter un type d'inventaire

Dans Systems Manager Inventory, les groupes vous permettent de voir rapidement un décompte des nœuds gérés qui sont ou ne sont pas configurés 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 nœuds gérés configurés pour collecter les types d'inventaire suivants :

- Nœud 1 : `AWS:Application`
- Nœud 2 : `AWS:File`
- Nœud 3 : `AWS:Application`, `AWS:File`
- Nœud 4 : `AWS:Network`

Vous pouvez exécuter la commande suivante à partir de la AWS CLI pour voir combien de nœuds sont configurés pour collecter les types d'inventaire `AWS:Application` et `AWS:File inventory`. La réponse renvoie également un décompte des nœuds qui ne sont pas configurés 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 de la commande montre qu'un seul nœud géré est configuré 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. De plus, vous ne pouvez pas explorer en détail les résultats pour voir les ID des nœuds qui sont ou non configurés 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 quels nœuds gérés sont ou non configurés 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 nœuds qui sont ou ne sont pas configurés 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 telles que les suivantes.

```
{
   "Entities":[
      {
         "Data":{
            "ApplicationORFile":{
               "Content":[
                  {
                     "notMatchingCount":"1"
                  },
                  {
                     "matchingCount":"3"
                  }
               ]
            }
         }
      }
   ]
}
```

Exemple 2

Cet exemple regroupe un décompte des nœuds qui sont ou ne sont pas configurés 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 telles que les 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 affecter toutes les métadonnées souhaitées à vos nœuds en créant un inventaire personnalisé Inventory AWS Systems Manager. 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 que nœuds gérés par 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 nœud en tant que métadonnées du nœud. 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 un maximum de 20 types d'inventaires personnalisés par Compte AWS.

Pour affecter un inventaire personnalisé à une nœud, vous pouvez utiliser l'action d'API Systems Manager PutInventory, comme décrit dans Démonstration : Affecter des métadonnées d'inventaire

personnalisé à un nœud géré. Sinon, vous pouvez créer un fichier JSON d'inventaire personnalisé et le charger sur le nœud. 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`, `TypeName` et `Content`, mais vous pouvez définir les informations dans ces sections.

```
{
    "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"
      }
}
```

La valeur de `TypeName` est limitée à 100 caractères. La valeur `TypeName` doit également commencer par le mot `Custom` en majuscule. Par exemple, `Custom:PuppetModuleInfo`. Par conséquent, les exemples suivants entraîneraient une exception : `CUSTOM:PuppetModuleInfo`, `custom:PuppetModuleInfo`.

La section `Content` inclut des attributs et des *données*. Ces éléments ne sont pas sensibles à la casse. Par contre, si vous définissez un attribut (par exemple: « `Vendor` » : « DELL »), vous devez faire référence à cet attribut de manière cohérente dans vos fichiers d'inventaire personnalisés. Si vous spécifiez « `Vendor` » : « DELL » (avec un « F » majuscule dans `vendor`) dans un fichier, puis « `vendor` » : « DELL » (avec un « f » minuscule dans `vendor`) dans un autre fichier, le système renvoie une erreur.

> **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 le nœud. Le tableau suivant montre l'emplacement où les fichiers JSON d'inventaires personnalisés doivent être stockés sur le nœud.

Système d'exploitation	Chemin
Linux	/var/lib/amazon/ssm/*node-id*/inventory/custom
macOS	`/opt/aws/ssm/data/` *node-id*`/inventory/custom`
Windows	%SystemDrive%\ProgramData\Amazon\SSM\InstanceData*node-id*\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'opération d'API DeleteInventory pour supprimer un type d'inventaire personnalisé et les données associées à celui-ci. Vous appelez la commande delete-inventory à l'aide de l'AWS Command Line Interface (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 sélectionnez cette option, le système désactive la version actuelle, supprime toutes les données qui lui sont associées et ignore toutes les nouvelles données si la version est antérieure ou égale à la version désactivée. Vous pouvez réactiver ce type d'inventaire en appelant l'action PutInventory 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 Command Line Interface (AWS CLI).

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l'AWS CLI.
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 retourne des informations telles que les suivantes.

```
{
   "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.

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. C'est pourquoi TotalCount et Remaining Count restent inchangés, 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 retourne des informations telles que les suivantes.

```
{
   "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 Systems Manager Inventory.

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 retourne des informations telles que les suivantes.

```
{
   "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 consulter 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 retourne des informations telles que les suivantes.

```
{
   "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 `describe-inventory-deletions` AWS CLI. 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 retourne des informations telles que les suivantes.

   ```
   {"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 est réussie, `LastStatusMessage` indique : Deletion is successful (Suppression réussie).

```
{"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 affecté l'inventaire Custom:RackSpace à trois nœuds. 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 personnalisé RackSpace 1

```
{
   "CaptureTime":"2018-02-19T10:48:55Z",
   "TypeName":"CustomType:RackSpace",
   "InstanceId":"i-1234567890",
   "SchemaVersion":"1.0"   "Content":[
      {
         content of custom type omitted
      }
   ]
}
```

Inventaire personnalisé RackSpace 2

```
{
   "CaptureTime":"2018-02-19T10:48:55Z",
   "TypeName":"CustomType:RackSpace",
   "InstanceId":"i-1234567891",
   "SchemaVersion":"1.0"   "Content":[
      {
         content of custom type omitted
      }
   ]
}
```

Inventaire personnalisé RackSpace 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 retourne des informations telles que les suivantes.

```
{
   "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. C'est pourquoi `TotalCount` et `RemainingCount` restent inchangés, 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 retourne des informations telles que les suivantes.

```
{
   "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 chaque fois qu'un utilisateur supprime l'inventaire personnalisé. EventBridge propose trois types d'événements pour les opérations de suppression d'inventaires personnalisés :

- Delete action for an instance (Action de suppression pour une instance) : événement créé si l'inventaire personnalisé pour un nœud géré spécifique a été correctement supprimé ou non.
- Delete action summary (Récapitulatif d'action de suppression) : récapitulatif de l'action de suppression.

- Warning for turned off custom inventory type (Avertissement pour le type d'inventaire personnalisé désactivé) : événement d'avertissement créé si un utilisateur a appelé l'opération d'API PutInventory pour une version de type d'inventaire personnalisé qui a été précédemment désactivée.

Vous trouverez ci-dessous 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 règle EventBridge 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 en savoir plus, consultez Mise en route dans le Guide du développeur Amazon Simple Notification Service.

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.
3. Choisissez Create rule.
4. 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.
5. Pour Event bus (Bus d'événement), sélectionnez le bus d'événement que vous souhaitez associer à cette règle. Si vous souhaitez que cette règle s'applique aux événements correspondants provenant de votre propre Compte AWS, sélectionnez défaut. Lorsqu'un Service AWS de votre compte émet un événement, il accède toujours au bus d'événement par défaut de votre compte.
6. Pour Rule type (Type de règle), choisissez Rule with an event pattern (Règle avec un modèle d'événement).
7. Choisissez Next (Suivant).
8. Pour Event source (Origine de l'événement), choisissez events or EventBridge partner events (Événements AWS ou événements partenaires EventBridge).
9. Dans la section Event pattern (Modèle d'événement), choisissez Event pattern form (Modèle d'événement).
10. Pour Event source (Origine de l'événement), choisissez AWSservices (Services).
11. Pour le AWS service choisissez Systems Manager.
12. Pour Type d'événement, sélectionnez Inventory.
13. Pour Specific detail type(s) (Type(s) spécifiques), choisissez Inventory Resource State Change (Changements d'état de la ressource Inventaire).
14. Choisissez Next (Suivant).
15. Pour Types de cibles, choisissez service AWS.
16. Pour Target (Cible), sélectionnez SNS topic (Rubrique SNS), puis sélectionnez votre rubrique depuis la liste Topic (Rubrique).

17. Dans la section Additional settings (Réglages supplémentaires), pour Configure target input (Configurer l'entrée cible), vérifiez que Matched event (Événement jumelé) est sélectionné.
18. Choisissez Next (Suivant).
19. (Facultatif) Saisissez une ou plusieurs balises pour la règle. Pour plus d'informations, consultez Balisage de vos ressources Amazon EventBridge dans le Guide de l'utilisateur Amazon EventBridge.
20. Choisissez Next (Suivant).
21. Consultez les détails de la règle et choisissez Create rule (Créer une règle).

Affichage de l'historique d'inventaire et suivi des modifications

Vous pouvez consulter l'historique d'inventaire AWS Systems Manager et le suivi des modifications pour l'ensemble de vos nœuds gérés avec 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 l'historique d'inventaire et le suivi des modifications, vous devez activer les ressources suivantes dans AWS Config :

- SSM:ManagedInstanceInventory
- SSM:PatchCompliance
- SSM:AssociationCompliance
- SSM:FileData

> **Note**
>
> Veuillez noter les informations suivantes, qui sont importantes pour l'historique Inventory et le suivi des modifications :
>
> - Si vous utilisez AWS Config pour suivre les modifications apportées à votre système, Systems Manager Inventory doit être configuré de sorte à collecter des métadonnées `AWS:File` et vous permettre ainsi d'afficher les modifications apportées aux fichiers dans AWS Config (`SSM:FileData`). Si ce n'est pas le cas, alors AWS Config ne suit pas les modifications apportées aux fichiers sur votre système.

- En activant SSM:PatchCompliance et SSM:AssociationCompliance, vous pouvez consulter l'historique de conformité et le suivi des modifications des associations State Manager et des correctifs Patch Manager de Systems Manager. Pour plus d'informations sur la gestion de la conformité pour ces ressources, consultez Utilisation du service Conformité.

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 Command Line Interface (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 Guide du développeur AWS Config. 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 une autorisation AWS Config aux ressources Systems Manager. Vous pouvez attacher la politique gérée `AWS_ConfigRole` au rôle IAM que vous attribuez à AWS Config. Pour plus d'informations sur ce rôle, consultez la rubrique Politique gérée par AWS : AWS_ConfigRole dans le Guide du développeur AWS Config. Pour savoir comment créer un rôle IAM et attribuer la politique gérée `AWS_ConfigRole` à ce rôle, reportez-vous à Création d'un rôle pour déléguer des autorisations à un Service AWS dans le Guide de l'utilisateur IAM.

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 Command Line Interface (AWS CLI).

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l'AWS CLI.
2. Copiez et collez l'exemple JSON suivant dans un fichier texte simple et enregistrez-le sous le nom recordingGroup.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 l'historique et le suivi des modifications configurés, vous pouvez explorer en détail l'historique pour rechercher un nœud géré spécifique en sélectionnant le bouton AWS Config dans la console Systems Manager. Vous pouvez accéder au bouton AWS Config à partir de la page Managed Instances (Instances gérées) ou de la page Inventory (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.

Arrêt de la collecte des données et suppression des données d'inventaire

Si vous ne souhaitez plus utiliser l' AWS Systems Manager inventaire pour afficher les métadonnées relatives à vos AWS ressources, vous pouvez arrêter la collecte de données et supprimer les données déjà collectées. Cette section comprend les informations suivantes.

Rubriques

- Arrêt de la collecte des données
- Suppression d'une synchronisation de données de ressources Inventory

Arrêt de la collecte des données

Lorsque System Manager est configuré initialement pour collecter des données d'inventaires, le système crée uneState Manager association définissant la planification ainsi que les ressources à partir desquelles les métadonnées seront collectées. Vous pouvez arrêter la collecte de données en supprimant des associations State Manager qui utilisent le document `AWS-GatherSoftwareInventory`.

Pour supprimer une association Inventory

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez State Manager.
3. Sélectionnez une association qui utilise le document `AWS-GatherSoftwareInventory`, puis sélectionnez Supprimer.
4. Répétez l'étape trois pour toutes les autres associations qui utilisent le document `AWS-GatherSoftwareInventory`.

Suppression d'une synchronisation de données de ressources Inventory

Si vous ne souhaitez plus utiliser l' AWS Systems Manager inventaire pour afficher les métadonnées relatives à vos AWS ressources, nous vous recommandons également de supprimer les synchronisations des données de ressources utilisées pour la collecte des données d'inventaire.

Pour supprimer une synchronisation de données de ressources Inventory

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le volet de navigation, sélectionnez Inventory.
3. Sélectionnez Resource Data Syncs (Synchronisations des données de ressource).
4. Dans la liste, sélectionnez une synchronisation.

> **Important**
>
> Veillez à bien choisir la synchronisation utilisée pour Inventory. Systems Manager prend en charge la synchronisation de données de ressources pour plusieurs fonctionnalités. Si

vous choisissez la mauvaise synchronisation, vous risquez de perturber l'agrégation des données pour Systems Manager Explorer ou Systems Manager Compliance.

5. Choisissez Delete (Supprimer)
6. Répétez ces étapes pour les autres synchronisations de données de ressources à supprimer.
7. Supprimez le compartiment Amazon Simple Storage Service (Amazon S3) dans lequel les données ont été enregistrées. Pour obtenir des informations sur la suppression d'un compartiment Amazon S3, veuillez consulter Deleting a bucket (Suppression d'un compartiment).

Procédures Systems Manager Invetory

Utilisez les démonstrations suivantes pour collecter et gérer les données d'inventaire à l'aide d'Inventory AWS Systems Manager. Nous vous recommandons de réaliser ces procédures pas à pas avec les nœuds gérés dans un environnement de test au préalable.

Avant de commencer

Avant de commencer ces procédures, exécutez les tâches suivantes :

- Mettez à jour AWS Systems Manager SSM Agent sur les nœuds à inventorier. En exécutant la dernière version de SSM Agent, 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'SSM Agent à l'aide de State Manager, consultez Démonstration : Mise à jour automatique de l'SSM Agent (CLI).
- Vérifiez que vous avez satisfait la configuration requise pour vos instances Amazon Elastic Compute Cloud (Amazon EC2) et vos machines non EC2 dans un environnement hybride et multicloud. Pour plus d'informations, consultez Con AWS Systems Manager figuration.
- (Facultatif) Créez un fichier JSON pour collecter l'inventaire personnalisé. Pour de plus amples informations, veuillez consulter Utilisation de l'inventaire personnalisé.

Table des matières

Démonstration : Affecter des métadonnées d'inventaire personnalisé à un nœud géré

La procédure suivante vous guide tout au long du processus d'utilisation de l'opération d'API AWS Systems Manager PutInventory pour attribuer des métadonnées d'inventaire personnalisé à un nœud géré. Cet exemple attribue les informations sur les emplacements des racks à un nœud. Pour plus d'informations sur l'inventaire personnalisé, consultez Utilisation de l'inventaire personnalisé.

Pour affecter des métadonnées d'inventaire personnalisé à un nœud

1. Si vous ne l'avez pas déjà fait, installez et configurez l'AWS Command Line Interface (AWS CLI).

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l'AWS CLI.
2. Exécutez la commande suivante pour affecter les informations de localisation de rack à un nœud.

   Linux

   ```
   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"}]'
   ```

   Windows

   ```
   aws ssm put-inventory --instance-id "ID" --items
    "TypeName=Custom:RackInfo,SchemaVersion=1.0,CaptureTime=2021-05-22T10:01:01Z,Content=[{Rac
    B/Row C/Rack D/Shelf F'}]"
   ```
3. Exécutez la commande suivante pour afficher les entrées de l'inventaire personnalisé de ce nœud.

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

Démonstration : Configurer vos nœuds gérés 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 AWS Systems Manager pour collecter les métadonnées de vos nœuds gérés. Lorsque vous configurez la collecte de données d'inventaire, vous commencez par créer une association Systems Manager State Manager. Systems Manager collecte les données d'inventaire lorsque l'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 Systems Manager Run Command, le système renvoie l'erreur suivante :

`The aws:softwareInventory plugin can only be invoked via ssm-associate.`

> **Note**
>
> Un nœud ne peut avoir qu'une association d'inventaire configurée en même temps. Si vous configurez un nœud 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 nœuds gérés pour l'inventaire (CLI)

Vous pouvez configurer rapidement tous les nœuds gérés de votre région Compte AWS et de la région actuelle pour collecter des 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 de tous les nœuds gérés dans votre région Compte AWS et dans la région actuelle (CLI)

1. Installez et configurez le AWS Command Line Interface (AWS CLI), si ce n'est pas déjà fait.

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l' AWS CLI.

2. Exécutez la commande suivante.

   Linux & macOS

   ```
   aws ssm create-association \
   --name AWS-GatherSoftwareInventory \
   --targets Key=InstanceIds,Values=* \
   --schedule-expression "rate(1 day)" \
   --parameters
    applications=Enabled,awsComponents=Enabled,customInventory=Enabled,instanceDetailedInfo
   ```

   Windows

   ```
   aws ssm create-association ^
   --name AWS-GatherSoftwareInventory ^
   --targets Key=InstanceIds,Values=* ^
   --schedule-expression "rate(1 day)" ^
   ```

```
--parameters
 applications=Enabled,awsComponents=Enabled,customInventory=Enabled,instanceDetailedInfo
```

> **Note**
>
> Cette commande n'autorise pas Inventory à collecter des métadonnées pour le registre Windows ou les fichiers. Pour effectuer l'inventaire de ces types de données, utilisez la procédure suivante.

Configuration manuelle de l'inventaire sur vos nœuds gérés (CLI)

Utilisez la procédure suivante pour configurer manuellement l' AWS Systems Manager inventaire sur vos nœuds gérés à l'aide d'identifiants ou de balises de nœud.

Pour configurer manuellement vos nœuds gérés pour l'inventaire (CLI)

1. Installez et configurez le AWS Command Line Interface (AWS CLI), si ce n'est pas déjà fait.

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l' AWS CLI.

2. Exécutez la commande suivante pour créer une association State Manager qui exécute Systems Manager Inventory sur le nœud. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations. 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'un nœud.

   Linux & macOS

   ```
   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 de nœuds en utilisant le paramètre `Targets` avec les balises EC2. Consultez l'exemple suivant.

Linux & macOS

```
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 un nœud 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.

Linux & macOS

```
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"
        }
]
```

```
}
```

Démonstration : utiliser la synchronisation de données de ressources pour regrouper les données d'inventaire

La procédure pas à pas suivante décrit comment créer une configuration de synchronisation des données de ressources pour AWS Systems Manager Inventory à l'aide du AWS Command Line Interface (AWS CLI). Une synchronisation de données de ressources transfère automatiquement les données d'inventaire collectées depuis tous vos nœuds gérés vers un compartiment Amazon Simple Storage Service (Amazon S3) central. La synchronisation met automatiquement à jour les données dans le compartiment Amazon S3 central lors de la découverte de nouvelles données d'inventaire.

Cette procédure pas à pas décrit également comment utiliser Amazon Athena et QuickSight Amazon pour interroger et analyser les données agrégées. Pour plus d'informations sur la création d'une synchronisation des données de ressources à l'aide de Systems Manager dans le AWS Management Console, voirConfiguration de la synchronisation de données de ressource pour Inventory. Pour plus d'informations sur l'interrogation de l'inventaire à partir de plusieurs comptes Régions AWS et à l'aide de Systems Manager dans le AWS Management Console, voirInterrogation des données d'inventaire à partir de plusieurs régions et comptes.

> **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 à ce sujet AWS KMS, consultez le Guide AWS Key Management Service du développeur.

Avant de commencer

Vérifiez ou effectuez les tâches suivantes avant de commencer la démonstration de cette section :

- Collectez les données d'inventaire de vos nœuds gérés. Dans le cadre des QuickSight sections Amazon Athena et Amazon de cette procédure pas à pas, nous vous recommandons de collecter les données de l'application. Pour plus d'informations sur la façon de collecter des données d'inventaire, consultez Configuration de la collecte d'inventaire ou Démonstration : Configurer vos nœuds gérés pour l'inventaire à l'aide de l'interface de ligne de commande.

- (Facultatif) Si les données d'inventaire sont stockées dans un bucket Amazon Simple Storage Service (Amazon S3) qui AWS Key Management Service utilise le chiffrement AWS KMS(), vous devez également configurer votre compte IAM et `Amazon-GlueServiceRoleForSSM` le rôle de service pour le chiffrement. AWS KMS Si vous ne configurez pas votre compte IAM et ce rôle, Systems Manager affiche `Cannot load Glue tables` lorsque vous sélectionnez l'onglet Vue détaillée sur la console. Pour plus d'informations, consultez (Facultatif) Configurer les autorisations d'affichage des données AWS KMS chiffrées.
- (Facultatif) Si vous souhaitez chiffrer la synchronisation des données des ressources en utilisant AWS KMS, vous devez soit créer une nouvelle clé incluant la politique suivante, soit mettre à jour une clé existante et y ajouter cette politique.

```
{
    "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:us-east-2:123456789012:key/KMS_key_id",
            "Condition": {
                "StringLike": {
                    "aws:SourceAccount": "123456789012"
                },
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:ssm:*:123456789012:resource-data-sync/
*"
                }
            }
        }
    ]
}
```

Pour créer une synchronisation des données de ressource pour l'inventaire

1. Ouvrez la console Amazon S3 sur https://console.aws.amazon.com/s3/.
2. Créez un compartiment pour stocker vos données d'inventaire agrégées. Pour plus d'informations, consultez Création d'un compartiment dans le Guide de l'utilisateur d'Amazon Simple Storage Service. Notez le nom du bucket et l' Région AWS endroit où vous l'avez créé.
3. Après avoir créé le compartiment, sélectionnez l'onglet Autorisations, puis sélectionnez Politique de compartiment.
4. Copiez et collez la politique de compartiment suivante dans l'éditeur de politique. Remplacez DOC-EXAMPLE-BUCKET et *account-id* par le nom du compartiment Amazon S3 que vous avez créé et un identifiant valide. Compte AWS Si vous ajoutez plusieurs comptes, ajoutez une chaîne de conditions et un ARN supplémentaires pour chaque compte. Supprimez les espaces réservés supplémentaires de l'exemple lors de l'ajout d'un compte. Vous avez également la possibilité de remplacer *bucket-prefix* par le nom d'un préfixe Amazon S3 (sous-répertoire). Si vous n'avez pas créé de préfixe, supprimez *bucket-prefix/* de l'ARN dans la politique.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "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",
          "aws:SourceAccount": [
            "account-id1",
            "account-id2",
            "account-id3",
            "account-id4"
          ]
        },
        "ArnLike": {
```

```
          "aws:SourceArn": [
            "arn:aws:ssm:*:account-id1:resource-data-sync/*",
            "arn:aws:ssm:*:account-id2:resource-data-sync/*",
            "arn:aws:ssm:*:account-id3:resource-data-sync/*",
            "arn:aws:ssm:*:account-id4:resource-data-sync/*"
          ]
        }
      }
    }
  ]
}
```

5. (Facultatif) Si vous souhaitez chiffrer la synchronisation, vous devez ajouter les conditions suivantes à la politique définie dans l'étape précédente. Ajoutez les dans la section `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"
```

   Voici un exemple :

```
"StringEquals": {
          "s3:x-amz-acl": "bucket-owner-full-control",
          "aws:SourceAccount": "account-id",
          "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. Installez et configurez le AWS Command Line Interface (AWS CLI), si ce n'est pas déjà fait.

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l' AWS CLI.

7. (Facultatif) Si vous souhaitez chiffrer la synchronisation, exécutez la commande suivante pour vérifier que la politique de compartiment applique les exigences relatives aux AWS KMS clés. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

Linux & macOS

```
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 Amazon S3 que vous avez créé au début de cette procédure. Cette commande crée une synchronisation à partir du compte auquel Région AWS vous êtes connecté.

> **Note**
>
> Si la synchronisation et le compartiment Amazon S3 cible sont localisés dans des régions différentes, vous pourriez être sujet à une tarification de transfert de données. Pour plus d'informations, consultez Tarification Amazon S3.

Linux & macOS

```
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 Amazon S3 de la région us-west-2.

Linux & macOS

```
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 souhaitez chiffrer la synchronisation à l'aide de AWS KMS, exécutez la commande suivante pour créer la synchronisation. Si vous chiffrez la synchronisation, la clé AWS KMS et le compartiment Amazon S3 doivent se trouver dans la même région.

Linux & macOS

```
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, examinez 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, consultez Qu'est-ce que Amazon Athena ? et Utilisation des données dans le Guide de l'utilisateur d'Amazon Athena.

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 sélectionnez 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 sélectionnez Exécuter la requête. Remplacez *DOC-EXAMPLE-BUCKET et bucket_prefix par le nom et le préfixe* de la cible Amazon S3.

   ```
   CREATE EXTERNAL TABLE IF NOT EXISTS ssminventory.AWS_Application (
   Name string,
   ```

```
ResourceId 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 sélectionnez 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 ressources supplémentaires Régions AWS ou Comptes AWS, vous devez exécuter cette commande à nouveau pour mettre à jour les partitions. Il est probable que vous deviez également mettre à jour votre politique de compartiment Amazon S3.

5. Pour prévisualiser vos données, sélectionnez l'icône Aperçu située à côté de la table `AWS_Application`.

6. Copiez et collez la déclaration suivante dans l'éditeur de requête, puis sélectionnez 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 le nombre de versions différentes de`aws-cfn-bootstrap`, qui est une AWS application présente sur les instances Amazon Elastic Compute Cloud (Amazon EC2) pour LinuxmacOS, et. Windows Server

7. Copiez et collez individuellement les instructions suivantes dans l'éditeur de requêtes, remplacez ***DOC-EXAMPLE-BUCKET et bucket-prefix*** par des informations pour 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
```

Travailler avec les données d'Amazon QuickSight

La section suivante fournit une vue d'ensemble avec des liens permettant de créer une visualisation dans Amazon QuickSight.

Pour créer une visualisation dans Amazon QuickSight

1. Inscrivez-vous à Amazon, QuickSight puis connectez-vous à la QuickSight console.
2. Créez un ensemble de données depuis la table `AWS_Application` et toute autre table que vous avez créée. Pour de plus amples informations, consultez Création d'un ensemble de données à l'aide d'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 la section Travailler avec Amazon QuickSight Visuals.

Résolution des problèmes liés à Systems Manager Inventory

Cette rubrique inclut des informations sur la résolution d'erreurs ou de problèmes courants liés à l'inventaire AWS Systems Manager. Si vous avez des difficultés à consulter vos nœuds dans Systems Manager, consultez Résolution des problèmes de disponibilité des nœuds gérés.

Rubriques

- Erreur « Multtiple apply all associations with document 'AWS-GatherSoftwareInventory' are not supported »
- Inventory ne quitte jamais le statut d'exécution « En attente »
- Le document AWS-ListWindowsInventory ne s'exécute pas
- La console n'affiche pas les éléments Tableau de bord d'inventaire | Vue détaillée | onglets Paramètres
- UnsupportedAgent
- Ignoré
- Échec
- Échec de conformité de l'inventaire pour une instance Amazon EC2
- L'objet du compartiment S3 contient d'anciennes données

Erreur « Multtiple apply all associations with document '**AWS-GatherSoftwareInventory**' are not supported »

L'erreur `Multiple apply all associations with document 'AWS-GatherSoftwareInventory' are not supported` signifie qu'une ou plusieurs Régions AWS dans lesquelles vous tentez de configurer une association Inventory pour tous les nœuds sont déjà configurées avec une association Inventory pour tous les nœuds. Si nécessaire, vous pouvez supprimer l'association Inventory existante pour tous les nœuds, puis en créer une nouvelle. Pour consulter les associations Inventory existantes, sélectionnez State Manager dans la

console Systems Manager, puis recherchez les associations qui utilisent le document SSM `AWS-GatherSoftwareInventory`. Si l'association Inventory existante pour tous les nœuds a été créée dans plusieurs régions et que vous voulez en créer une nouvelle, vous devez supprimer l'association existante dans chaque région où elle est présente.

Inventory ne quitte jamais le statut d'exécution « En attente »

Deux raisons prévalent au fait que la collecte de données d'inventaire ne quitte jamais le statut `Pending`.

- Aucun nœud dans la Région AWS sélectionnée :

  Si vous créez une association Inventory globale avec Systems Manager Quick Setup, le statut de l'association Inventory (document `AWS-GatherSoftwareInventory`) affiche `Pending` s'il n'y a aucun nœud disponible dans la région sélectionnée.

- Autorisations insuffisantes :

  Une association Inventory affiche `Pending` si un ou plusieurs nœuds n'ont pas l'autorisation d'exécuter Systems Manager Inventory. Vérifiez que le profil d'instance AWS Identity and Access Management (IAM) inclut la politique gérée AmazonSSMManagedInstanceCore. Pour obtenir des informations sur l'ajout de cette politique à un profil d'instance, veuillez consulter Configuration alternative pour les autorisations d'instance EC2.

  Le profil d'instance doit disposer au moins 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"
```

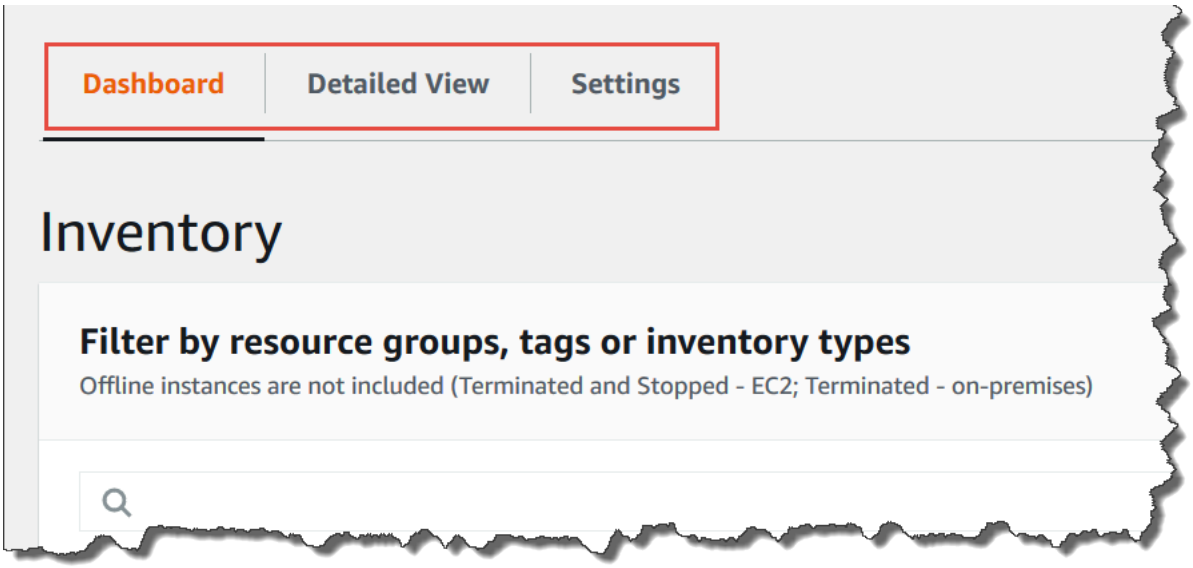
```
            ],
            "Resource": "*"
        }
    ]
}
```

Le document **AWS-ListWindowsInventory** ne s'exécute pas

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.

La console n'affiche pas les éléments Tableau de bord d'inventaire | Vue détaillée | onglets Paramètres

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 Inventory, 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.



UnsupportedAgent

Si le statut détaillé d'une association d'inventaire indique UnsupportedAgent (Agent non pris en charge), et que Association status (Statut d'association) indique Failed (Échec), la version de AWS Systems Manager SSM Agent sur le nœud géré est incorrecte. Pour créer une association d'inventaire globale (pour tous les nœuds d'inventaire de votre Compte AWS) par exemple, vous

devez utiliser l'SSM Agent version 2.0.790.0 ou ultérieure. Vous pouvez consulter la version de l'agent en cours d'exécution sur chacune de vos nœuds sur la page Managed Instances (Instances gérées) dans la colonne Agent version (Version d'agent). Pour plus d'informations sur la mise à jour de l'SSM Agent sur vos nœuds, consultez Mise à jour de SSM Agent à l'aide de Run Command.

Ignoré

Si le statut de l'association d'inventaire pour un nœud indique Skipped (Ignoré), cela signifie que vous avez créé une association d'inventaire global (pour collecter l'inventaire de tous les nœuds), mais qu'une association d'inventaire était déjà attribuée au nœud ignoré. L'association d'inventaire global n'a pas été attribuée à ce nœud, et aucun inventaire n'a été collecté par l'association d'inventaire global. Toutefois, le nœud continue de signaler des données d'inventaire lorsque l'association d'inventaire existant s'exécute.

Si vous ne voulez pas que le nœud soit ignoré par l'association d'inventaire global, vous devez supprimer l'association d'inventaire existante. Pour consulter les associations Inventory existantes, sélectionnez State Manager dans la console Systems Manager, puis recherchez les associations qui utilisent le document SSM `AWS-GatherSoftwareInventory`.

Échec

Si le statut de l'association d'inventaire pour un nœud indique Failed (Échec), cela peut signifier que plusieurs plusieurs associations d'inventaire lui sont affectées. Une seule association d'inventaire peut être attribuée à un nœud à la fois. Une association Inventory utilise le document AWS Systems Manager (document SSM) `AWS-GatherSoftwareInventory`. Vous pouvez exécuter la commande suivante à l'aide de l'AWS Command Line Interface (AWS CLI) pour afficher la liste des associations pour un nœud.

```
aws ssm describe-instance-associations-status
                --instance-id instance ID
```

Échec de conformité de l'inventaire pour une instance Amazon EC2

L'échec de la conformité de l'inventaire pour une instance Amazon Elastic Compute Cloud (Amazon EC2) est possible si vous attribuez plusieurs associations d'inventaire à l'instance.

Pour résoudre ce problème, passez à la suppression d'une ou plusieurs associations d'inventaire attribuées à l'instance. Pour de plus amples informations, consultez Création d'une association.

> **Note**
>
> Vous devez connaître le comportement suivant en cas de création d'associations d'inventaire multiples pour un nœud géré.
>
> - Chaque nœud peut se voir affecter une association d'inventaire ciblant tous les nœuds (--targets "Key=InstanceIds,Values=*").
> - Chaque nœud peut également être doté d'une association spécifique utilisant des paires clé-valeur de balise ou un groupe de ressources AWS.
> - Si plusieurs associations d'inventaire sont affectées à un nœud, l'association qui ne s'est pas exécutée présente le statut Skipped (Ignoré). L'association qui s'est exécutée la plus récemment affiche le statut réel de l'association d'inventaire.
> - Si un nœud est doté de plusieurs associations d'inventaire et utilise une paire clé-valeur de balise, le conflit de balises empêche l'exécution de ces associations d'inventaire sur le nœud. L'association s'exécute normalement sur les nœuds exempts du conflit clé-valeur de balise.

L'objet du compartiment S3 contient d'anciennes données

Les données contenues dans l'objet du compartiment Amazon S3 sont mises à jour lorsque l'association d'inventaire est réussie et que de nouvelles données sont découvertes. L'objet du compartiment Amazon S3 est mis à jour pour chaque nœud lorsque l'association s'exécute et échoue, mais les données contenues dans l'objet ne sont pas mises à jour dans ce cas. Les données contenues dans l'objet du compartiment Amazon S3 ne seront mises à jour que lorsque l'association s'exécutera correctement. Lorsque l'association d'inventaire échoue, vous verrez d'anciennes données dans l'objet du compartiment Amazon S3.

AWS Systems Manager Activations hybrides

Pour configurer des machines non EC2 afin de les utiliser AWS Systems Manager dans un environnement hybride et multicloud, vous devez créer une activation hybride. Les types de machines non EC2 pris en charge en tant que nœuds gérés sont les suivants :

- Serveurs sur votre propre site (serveurs sur site)
- AWS IoT Greengrass appareils principaux
- AWS IoT et appareils non AWS périphériques

- Machines virtuelles (VM), y compris les VM dans d'autres environnements cloud

Lorsque vous exécutez la commande create-activation pour lancer un processus d'activation hybride, vous recevez un code d'activation et un ID dans la réponse de la commande. Vous incluez ensuite le code d'activation et l'ID dans la commande d'installation de SSM Agent sur l'appareil, comme décrit à l'étape 3 de Utilisation de Systems Manager dans des environnements hybrides et multicloud. Ce processus d'activation s'applique à tous les types de machines non EC2, à l'exception des appareils AWS IoT Greengrass principaux. Pour plus d'informations sur la configuration des périphériques AWS IoT Greengrass principaux pour Systems Manager, consultezGestion des appareils de pointe avec Systems Manager.

> **Note**
>
> Aucune prise en charge n'est actuellement fournie pour les appareils macOS non EC2.

À propos des niveaux d'instances Systems Manager

AWS Systems Manager propose un niveau d'instances standard et un niveau d'instances avancées. Les deux prennent en charge les nœuds gérés dans votre environnement hybride et multicloud. Le niveau d'instances standard vous permet d'enregistrer un maximum de 1 000 machines par machine. Compte AWS Région AWS Si vous avez besoin d'enregistrer plus de 1 000 machines dans un seul compte et une seule région, utilisez le niveau d'instances avancées. Le niveau d'instances avancées vous permet de créer autant de nœuds gérés que vous le souhaitez. Tous les nœuds gérés configurés pour Systems Manager sont facturés sur une pay-per-use base. Pour plus d'informations sur l'activation des instances avancées, consultez Activation du niveau d'instances avancées. Pour plus d'informations sur la tarification, consultez Tarification AWS Systems Manager.

> **Note**
>
> - Les instances avancées vous permettent également de vous connecter à vos nœuds non EC2 dans un environnement hybride et multicloud en utilisant. AWS Systems Manager Session Manager Session Managerfournit un accès shell interactif à vos instances. Pour plus d'informations, consultez AWS Systems Manager Session Manager.
> - 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).

- Pour corriger les applications publiées par Microsoft sur des instances de machines virtuelles (VM) sur site, activez le niveau d'instances avancées. L'utilisation du niveau d'instance avancé est facturée. La correction d'applications publiées par Microsoft sur des instances Amazon Elastic Compute Cloud (Amazon EC2) n'induit aucuns frais supplémentaires. Pour plus d'informations, voir À propos de la correction d'applications publiées par Microsoft sur Windows Server.

AWS Systems Manager Session Manager

Session Managerest une AWS Systems Manager fonctionnalité entièrement gérée. Avec Session Manager, vous pouvez gérer vos instances Amazon Elastic Compute Cloud (Amazon EC2), appareils de périphérie, serveurs sur site et machines virtuelles. Vous pouvez utiliser soit un shell interactif basé sur un navigateur en un clic, soit le AWS Command Line Interface ().AWS CLISession Managerfournit une gestion des nœuds sécurisée et vérifiable sans qu'il soit nécessaire d'ouvrir les ports entrants, de gérer les hôtes Bastion ou de gérer les clés SSH. Session Managervous permet également de vous conformer aux politiques d'entreprise qui exigent un accès contrôlé aux nœuds gérés, des pratiques de sécurité strictes et des journaux entièrement vérifiables avec les détails d'accès aux nœuds, tout en fournissant aux utilisateurs finaux un accès multiplateforme en un clic à vos nœuds gérés. Pour vos premiers pas dans Session Manager, ouvrez Systems Manager console. Dans le panneau de navigation, sélectionnez Session Manager.

Comment mon organisation peut-elle tirer parti de Session Manager ?

Session Manager offre les avantages suivants :

- Contrôle centralisé des accès aux nœuds gérés à l'aide de politiques IAM

  Les administrateurs disposent d'un point central pour accorder et révoquer les accès aux nœuds gérés. En utilisant uniquement des politiques AWS Identity and Access Management (IAM), vous pouvez contrôler quels utilisateurs ou groupes individuels de votre organisation peuvent utiliser Session Manager et à quels nœuds gérés ils peuvent accéder.
- Aucun port entrant ouvert et aucune nécessité d'ouvrir des hôtes bastion ou des clés SSH

  En laissant les ports SSH et les ports PowerShell distants ouverts sur vos nœuds gérés, vous augmentez considérablement le risque que des entités y exécutent des commandes malveillantes ou non autorisées. Session Manager vous aide à renforcer votre niveau de sécurité en vous

permettant de fermer ces ports entrants, et d'éviter ainsi de gérer les clés SSH et les certificats, les hôtes bastion ainsi que les zones de reroutage.

- Accès en un clic aux nœuds gérés depuis la console et la CLI

  À l'aide de la AWS Systems Manager console ou de la console Amazon EC2, vous pouvez démarrer une session en un seul clic. À l'aide du AWS CLI, vous pouvez également démarrer une session qui exécute une seule commande ou une séquence de commandes. Étant donné que les autorisations d'accès aux nœuds gérés sont fournies via des politiques IAM et non des clés SSH ou autres mécanismes, le temps de connexion est considérablement réduit.

- Connexion à la fois aux instances Amazon EC2 et aux nœuds gérés non EC2 dans des environnements hybrides et multicloud

  Vous pouvez vous connecter à la fois aux instances Amazon Elastic Compute Cloud (Amazon EC2) et aux nœuds non EC2 de votre environnement hybride et multicloud.

  Pour vous connecter à des nœuds non EC2 à l'aide de Session Manager, activez d'abord le niveau d'instances avancées. L'utilisation du niveau d'instance avancé est payante. Cependant, il n'existe aucun frais supplémentaires pour la connexion aux instances EC2 à l'aide de Session Manager. Pour plus d'informations, consultez Configuration des niveaux d'instance.

- Réacheminement de port

  Redirigez n'importe quel port de votre nœud géré distant vers un port local sur un client. Après cela, connectez-vous au port local et accédez à l'application serveur qui s'exécute sur le nœud.

- Prise en charge multiplateforme de Windows, Linux et macOS

  Session Manager prend en charge Windows, Linux et macOS à partir d'un simple outil. Par exemple, vous n'avez pas besoin d'utiliser un client SSH pour les nœuds gérés Linux et macOS, ni une connexion RDP pour les nœuds gérés 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 nœuds gérés 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, intégrées aux Services AWS suivants :

- AWS CloudTrail— AWS CloudTrail capture les informations relatives aux appels d'Session ManagerAPI effectués dans votre compte Compte AWS et les écrit dans des fichiers journaux qui sont stockés dans un bucket Amazon Simple Storage Service (Amazon S3) que vous spécifiez. Un compartiment est utilisé pour tous les CloudTrail journaux de votre compte. Pour plus d'informations, consultez Journalisation des appels d' AWS Systems Manager API avec AWS CloudTrail.
- Amazon Simple Storage Service : vous pouvez choisir de stocker les données des journaux de session dans le compartiment Amazon S3 de votre choix à des fins de débogage et de résolution des problèmes. Les données des journaux peuvent être envoyées à votre compartiment Amazon S3 avec ou sans chiffrement à l'aide de votre AWS KMS key. Pour plus d'informations, consultez Journalisation des données de session avec Amazon S3 (console).
- Amazon CloudWatch Logs — CloudWatch Logs vous permet de surveiller, de stocker et d'accéder à des fichiers journaux provenant de différents sites Services AWS. Vous pouvez envoyer les données du journal de session à un groupe de CloudWatch journaux de journaux à des fins de débogage et de résolution des problèmes. Les données de journal peuvent être envoyées à votre groupe de journaux avec ou sans AWS KMS chiffrement à l'aide de votre clé KMS. Pour plus d'informations, consultez Enregistrement des données de session à l'aide d'Amazon CloudWatch Logs (console).
- Amazon EventBridge et Amazon Simple Notification Service : vous EventBridge permettent de définir des règles pour détecter les modifications apportées aux AWS ressources 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 lancer d'autres réponses. Pour plus d'informations, consultez Surveillance de l'activité des sessions à l'aide d'Amazon EventBridge (console).

> **Note**
>
> La journalisation n'est pas disponible pour les sessions Session Manager qui se connectent via le réacheminement de port ou SSH. Cela est dû au fait que SSH chiffre toutes les données de session et que Session Manager sert uniquement de tunnel pour les connexions SSH.

À qui est destiné Session Manager ?

- Tout AWS client qui souhaite améliorer sa sécurité et sa posture d'audit, réduire les frais opérationnels en centralisant le contrôle d'accès sur les nœuds gérés et réduire l'accès aux nœuds entrants.
- Les experts en sécurité de l'information qui souhaitent surveiller et suivre l'accès aux nœuds et leur activité, fermer les ports entrants sur les nœuds gérés ou autoriser les connexions à des nœuds gérés sans adresse IP publique.
- Administrateurs qui souhaitent accorder et révoquer des accès à partir d'un point central, et fournir aux utilisateurs une solution unique pour les nœuds gérés Linux, macOS et Windows Server.
- Les utilisateurs qui souhaitent se connecter à un nœud géré 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 de nœuds gérés Windows Server, Linux et macOS

  Session Manager vous permet d'établir des connexions sécurisées à vos instances Amazon Elastic Compute Cloud (EC2), appareils de périphérie, serveurs sur site et machines virtuelles. Pour obtenir une liste des types de systèmes d'exploitation pris en charge, consultez Configuration de Session Manager.

  > **Note**
  >
  > La prise en charge de Session Manager pour les machines sur site est assurée pour le niveau d'instances avancées uniquement. Pour plus d'informations, consultez Activation du niveau d'instances avancées.

- Accès aux fonctionnalités de Session Manager via la console, l'interface de ligne de commande et le kit SDK

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

  La console AWS Systems Manager inclut l'accès à toutes les fonctionnalités de Session Manager pour les administrateurs et les utilisateurs finaux. Vous pouvez effectuer n'importe quelle tâche liée à vos sessions via la console Systems Manager.

La console Amazon EC2 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, consulter la liste des sessions et y mettre fin définitivement en utilisant le AWS CLI.

> **Note**
>
> Pour utiliser les commandes AWS CLI d'exécution de session, vous devez utiliser la version 1.16.12 de la CLI (ou ultérieure) et vous devez avoir installé le Session Manager plug-in sur votre machine locale. Pour plus d'informations, veuillez consulter Installez le Session Manager plugin pour AWS CLI. Pour voir le plugin activéGitHub, consultez session-manager-plugin.

- Contrôle d'accès IAM

  Les politiques IAM vous permettent de contrôler quels membres de votre organisation peuvent démarrer des sessions sur les nœuds gérés et à quels nœuds ils peuvent accéder. Vous pouvez également fournir un accès temporaire à vos nœuds gérés. 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 rotation.

- Prise en charge de la capacité de journalisation et d'audit

  Session Managervous fournir des options pour auditer et enregistrer l'historique de vos sessions Compte AWS grâce à l'intégration à un certain nombre d'autres Services AWS. Pour plus d'informations, consultez Auditer l'activité de session et Activation et désactivation de la journalisation des activités de session.

- Profils de shell configurables

  Session Manager vous propose des options de configuration des préférences de session. Ces profils personnalisables vous permettent de définir des préférences telles que les préférences du shell, les variables d'environnement, les répertoires de travail et l'exécution de plusieurs commandes au démarrage d'une session.

- Prise en charge du chiffrement des données clés des clients

Vous pouvez configurer Session Manager pour chiffrer les journaux de données de session que vous envoyez à un bucket Amazon Simple Storage Service (Amazon S3) ou que vous diffusez vers CloudWatch un groupe de journaux de journaux. Vous pouvez également configurer Session Manager pour chiffrer davantage les données transmises entre les ordinateurs clients et vos nœuds gérés pendant vos sessions. Pour obtenir des informations, consultez Activation et désactivation de la journalisation des activités de session et Configurer les préférences de session.

- AWS PrivateLink prise en charge des nœuds gérés sans adresses IP publiques

  Vous pouvez également configurer des points de terminaison VPC pour Systems Manager afin de AWS PrivateLink sécuriser davantage vos sessions. AWS PrivateLink limite tout le trafic réseau entre vos nœuds gérés, Systems Manager et Amazon EC2 vers le réseau Amazon. Pour plus d'informations, consultez Améliorer la sécurité des instances EC2 en utilisant des points de terminaison VPC pour Systems Manager.

- Tunnelisation

  Dans une session, utilisez un document de type session AWS Systems Manager (SSM) pour canaliser le trafic, tel que le protocole HTTP ou un protocole personnalisé, entre un port local sur un ordinateur client et un port distant sur un nœud géré.

- 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 un nœud géré.

Qu'est-ce qu'une session ?

Une session est une connexion établie à un nœud géré en utilisant Session Manager. Les sessions sont basées sur un canal de communication bidirectionnel sécurisé entre le client (vous) et l'instance gérée à un nœud géré qui diffuse les entrées et les sorties pour les commandes. Le trafic entre un client et un nœud géré est chiffré via TLS 1.2, et les demandes de création de la connexion sont signées via Sigv4. Cette communication bidirectionnelle permet le bash interactif et PowerShell l'accès aux nœuds gérés. Vous pouvez également utiliser une clé AWS Key Management Service (AWS KMS) pour renforcer le chiffrement des 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 un nœud géré. 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 un nœud. À l'aide de la AWS Systems Manager console, de la console Amazon EC2 ou de la AWS CLI, John démarre une session le connectant au nœud géré, exécute des commandes sur le nœud nécessaire pour effectuer la tâche, puis met fin à la session.

Lorsque John envoie la première commande pour démarrer la session, le service Session Manager authentifie son ID, vérifie les autorisations qui lui ont été accordées par une politique IAM, vérifie les paramètres de configuration (par exemple, les limites autorisées pour les sessions), et envoie un message à l'SSM Agent 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'SSM Agent est chargé vers ce canal de communication et renvoyé vers son ordinateur local.

Rubriques

Configuration de Session Manager

Avant de vous connecter AWS Systems Manager Session Manager aux nœuds gérés de votre compte, suivez les étapes décrites dans les rubriques suivantes.

Rubriques

- Étape 8 : (Facultatif) Autoriser et contrôler les autorisations pour les connexions SSH via Session Manager

Étape 1 : Exécution des conditions Session Manager prérequises

Avant d'utiliser Session Manager, vérifiez que votre environnement respecte les conditions requises suivantes.

Conditions préalables requises Session Manager

Exigence	Description
Systèmes d'exploitation pris en charge	Session Manager prend en charge la connexion aux instances Amazon Elastic Compute Cloud (Amazon EC2), ainsi qu'aux machines non EC2 de votre environnement hybride et multicloud qui utilisent le niveau d'instances avancées. Session Manager prend en charge les versions de système d'exploitation suivantes : **Note** Session Manager prend en charge les instances EC2, les appareils de périphérie, les serveurs sur site et les machines virtuelles (VM) de votre environnement hybride et multicloud qui utilisent le niveau d'instances avancées. Pour de plus amples informations, sur les instances avancées, veuillez consulter Configuration des niveaux d'instance. Linux et macOS

Exigence	Description
	Session Managerprend en charge toutes les versions de Linux et macOS qui sont prises en charge par AWS Systems Manager. Pour plus d'informations, veuillez consulter Systèmes d'exploitation et types de machines pris en charge. Windows Session Manager prend en charge Windows Server 2012 via Windows Server 2022. **Note** Microsoft Windows Server 2016 Nano n'est pas pris en charge.

Exigence	Description
SSM Agent	Au minimum, AWS Systems Manager SSM Agent la version 2.3.68.0 ou ultérieure doit être installée sur les nœuds gérés auxquels vous souhaitez vous connecter via des sessions. Pour utiliser l'option permettant de chiffrer les données de session à l'aide d'une clé créée dans AWS Key Management Service (AWS KMS), la version 2.3.539.0 ou ultérieure de SSM Agent doit être installée sur le nœud géré. Pour utiliser des profils de shell dans une session, SSM Agent version 3.0.161.0 ou version ultérieure doit être installé sur le nœud géré. Pour démarrer une session de réacheminement de port Session Manager ou SSH, SSM Agent version 3.0.222.0 ou version ultérieure doit être installé sur le nœud géré. Pour diffuser des données de session à l'aide d'Amazon CloudWatch Logs, SSM Agent la version 3.0.284.0 ou ultérieure doit être installée sur le nœud géré. Pour en savoir plus sur la façon de déterminer le numéro de version exécuté sur une instance, consultez Vérification du numéro de version de l'SSM Agent. Pour plus d'informations sur l'installation manuelle ou la mise à jour automatique de SSM Agent, veuillez consulter Utilisation de l'option SSM Agent. À propos du compte ssm-user

Exigence	Description
	A partir de la version 2.3.50.0 de SSM Agent, l'agent crée un compte utilisateur sur le nœud géré, avec des autorisations racine ou administrateur, nommé `ssm-user`. (Sur les versions antérieures à 2.3.612.0, le compte est créé lorsque SSM Agent démarre ou redémarre . Sur la version 2.3.612.0, et ultérieure, `ssm-user` est créé la première fois qu'une session démarre sur le nœud géré.) Les sessions sont lancées à l'aide des informations d'identification administratives de ce compte utilisateur. Pour obtenir des informations sur la limitation du contrôle administratif pour ce compte, veuillez consulter Désactiver ou activer les autorisations administratives du compte ssm-user. ssm-user sur les contrôleurs de domaine Windows Server Depuis la version 2.3.612.0 de SSM Agent, le compte `ssm-user` n'est plus créé automatiq uement sur les nœuds gérés qui sont utilisés en tant que contrôleurs de domaine Windows Server. Pour utiliser Session Manager sur une machine Windows Server fonctionnant comme contrôleur de domaine, vous devez créer le compte `ssm-user` manuellement, s'il n'est pas déjà présent, et affecter des autorisations d'administrateur de domaine à l'utilisateur. Sur Windows Server, SSM Agent 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
Connectivité aux points de terminaison	Les nœuds gérés que vous connectez 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 Pour plus d'informations, consultez les rubriques suivantes : • Référence : ec2messages, ssmmessages et autres opérations d'API • Comment créer des points de terminaison VPC afin de pouvoir utiliser Systems Manager pour gérer des instances EC2 privées sans accès à Internet ? dans le AWS re:Post Knowledge Center. Vous pouvez également vous connecter aux points de terminaison requis à l'aide de points de terminaison d'interface. Pour plus d'informations, consultez Étape 6 (facultative) : Utiliser AWS PrivateLink pour configurer un point de terminaison de VPC pour Session Manager.

Exigence	Description
AWS CLI	(Facultatif) Si vous utilisez le AWS Command Line Interface (AWS CLI) pour démarrer vos sessions (au lieu d'utiliser la AWS Systems Manager console ou la console Amazon EC2), la version 1.16.12 ou ultérieure de la CLI doit être installée sur votre machine locale. Vous pouvez appeler `aws --version` pour vérifier la version. Si vous devez installer ou mettre à niveau la CLI, reportez-vous à la section Installation de la AWS Command Line Interface dans le guide de AWS Command Line Interface l'utilisateur. **Important** Une nouvelle version de SSM Agent est lancée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Le fait de ne pas utiliser la dernière version de l'agent peut empêcher votre nœud géré d'utiliser diverses capacités et fonctionnalités de Systems Manager. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir SSM Agent à jour sur vos machines. Pour plus d'informations, consultez Automatisation des mises à jour de l'SSM Agent. Abonnez-vous à la page des notes de SSM Agent publication GitHub pour recevoir des notifications

Exigence	Description
	concernant les SSM Agent mises à jour. En outre, pour utiliser la CLI afin de gérer vos nœuds avec Session Manager, vous devez d'abord installer le plugin Session Manager sur votre machine locale. Pour plus d'informations, veuillez consulter Installez le Session Manager plugin pour AWS CLI.
Activation du niveau d'instances avancées (environnements hybrides et multicloud)	Pour vous connecter à des machines non EC2 à l'aide deSession Manager, vous devez activer le niveau d'instances avancées dans Compte AWS et dans Région AWS lequel vous créez des activations hybrides pour enregistrer des machines non EC2 en tant que nœuds gérés. L'utilisation du niveau d'instance avancé est facturée. Pour plus d'informations sur le niveau d'instances avancées, consultez Configuration des niveaux d'instance.

Exigence	Description
Vérification des autorisations de fonction du service IAM (environnements hybrides et multicloud)	Les nœuds activés par hybride utilisent le rôle de service AWS Identity and Access Management (IAM) spécifié lors de l'activation hybride pour communiquer avec les opérations de l'API Systems Manager. Cette fonction du service doit contenir les autorisations requises pour se connecter à vos machines hybrides et multicloud à l'aide de Session Manager. Si votre rôle de service contient la politique AWS gérée `AmazonSSMManagedInstanceCore`, les autorisations requises pour Session Manager sont déjà fournies. Si vous constatez que la fonction de service ne contient pas les autorisations requises, vous devez désenregistrer l'instance gérée et l'enregistrer auprès d'une nouvelle activation hybride utilisant une fonction de service IAM avec les autorisations requises. Pour plus d'informations sur l'annulation de l'enregistrement d'instances gérées, consultez Annulation de l'enregistrement de nœuds gérés dans un environnement hybride et multicloud. Pour plus d'informations sur la création des politiques IAM avec des autorisations Session Manager, consultez Étape 2 : Vérification ou ajout d'autorisations d'instance pour Session Manager.

Étape 2 : vérifier ou ajouter des autorisations d'instance pour Session Manager

Par défaut, AWS Systems Manager n'est pas autorisé à effectuer des actions sur vos instances. Vous pouvez fournir des autorisations d'instance au niveau du compte à l'aide d'un rôle AWS Identity and Access Management (IAM), ou au niveau de l'instance à l'aide d'un profil d'instance.

Si votre cas d'utilisation le permet, nous vous recommandons d'accorder l'accès au niveau du compte à l'aide de la configuration de gestion des hôtes par défaut. Si vous avez déjà configuré la configuration de gestion des hôtes par défaut pour votre compte à l'aide de la politique `AmazonSSMManagedEC2InstanceDefaultPolicy`, vous pouvez passer à l'étape suivante. Pour plus d'informations sur la Configuration de gestion des hôtes par défaut, consultez Utilisation du paramètre de configuration de gestion d'hôte par défaut.

Vous pouvez également utiliser des profils d'instance pour fournir les autorisations requises à vos instances. Un profil d'instance transmet un rôle IAM à une instance Amazon EC2. Vous pouvez attacher un profil d'instance IAM à une instance Amazon EC2 lorsque vous la lancez ou à une instance préalablement lancée. Pour plus d'informations, consultez Utilisation de profils d'instance.

Pour les serveurs sur site ou les machines virtuelles, les autorisations sont fournies par la fonction du service IAM associé à l'activation hybride utilisée pour enregistrer vos serveurs sur site et machines virtuelles auprès de Systems Manager. Les serveurs sur site et les machines virtuelles n'utilisent pas les profils d'instance.

Si vous utilisez déjà d'autres fonctionnalités Systems Manager, comme Run Command 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 Amazon EC2. Si un profil d'instance contenant la politique gérée par AWS `AmazonSSMManagedInstanceCoreest` déjà attaché à vos instances, les autorisations requises pour Session Manager sont déjà fournies. Cela est également vrai si la fonction du service IAM utilisée dans votre activation hybride contient la politique gérée `AmazonSSMManagedInstanceCore`.

> **⚠ Important**
>
> Vous ne pouvez pas modifier la fonction du service IAM associée à une activation hybride. Si vous constatez que la fonction de service ne contient pas les autorisations requises, vous devez désenregistrer l'instance gérée et l'enregistrer auprès d'une nouvelle activation hybride utilisant une fonction de service avec les autorisations requises. Pour plus d'informations sur l'annulation de l'enregistrement d'instances gérées, consultez Annulation de l'enregistrement de nœuds gérés dans un environnement hybride et multicloud. Pour plus d'informations sur la création d'un rôle de service IAM pour les machines sur site, voir Créer le rôle de service IAM requis pour Systems Manager dans les environnements hybrides et multicloud.

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 politique personnalisée pour votre profil d'instance ou vous souhaitez utiliser le chiffrement Amazon Simple Storage Service (Amazon S3) AWS Key Management Service ou les options de chiffrement 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 :

- Intégration des autorisations pour les actions Session Manager dans un rôle IAM personnalisé

  Pour ajouter des autorisations pour Session Manager des actions à un rôle IAM existant qui ne repose pas sur la politique par défaut AWS fournie`AmazonSSMManagedInstanceCore`, suivez les étapes décrites dans. Ajout d'autorisations Session Manager à un rôle IAM existant
- Création d'un rôle IAM personnalisé contenant uniquement des autorisations Session Manager

  Pour créer un rôle IAM qui contient uniquement des autorisations relatives aux actions Session Manager, suivez les étapes décrites sur la page Création d'un rôle IAM personnalisé pour Session Manager.
- Création et utilisation d'un rôle IAM autorisant toutes les actions Systems Manager

  Pour créer un rôle IAM pour les instances gérées par Systems Manager qui utilise une politique par défaut fournie par AWS pour accorder toutes les autorisations de Systems Manager, suivez les étapes décrites dans Configurer les autorisations d'instance requises pour Systems Manager.

Rubriques

Ajout d'autorisations Session Manager à un rôle IAM existant

Utilisez la procédure suivante pour ajouter des autorisations Session Manager à un rôle (IAM) AWS Identity and Access Management existant. En ajoutant des autorisations à un rôle existant, vous pouvez améliorer la sécurité de votre environnement informatique sans avoir à utiliser la politique AWS `AmazonSSMManagedInstanceCore` pour les autorisations d'instance.

> **Note**
>
> Veuillez noter les informations suivantes :
>
> - Notez que cette procédure suppose que votre rôle existant comprend déjà d'autres autorisations `ssm` Systems Manager relatives aux actions auxquelles vous souhaitez accorder l'accès. Employée seule, cette politique ne s'avère pas suffisante pour utiliser Session Manager.
> - L'exemple de politique suivant inclut une action `s3:GetEncryptionConfiguration`. Cette action est requise si vous avez choisi l'option Appliquer le chiffrement du journal S3 dans les préférences de journalisation de Session Manager.

Pour ajouter des autorisations Session Manager à un rôle existant (console)

1. Connectez-vous à l'outil AWS Management Console, puis ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation, sélectionnez Rôles.
3. Choisissez le nom du rôle auquel vous souhaitez ajouter les autorisations.
4. Choisissez l'onglet Permissions (Autorisations).
5. Sélectionnez Ajouter des autorisations, puis Créer la politique en ligne.
6. Sélectionnez l'onglet JSON.
7. Remplacez le contenu de politique par défaut par le contenu suivant. Remplacez le *nom de clé* par l'Amazon Resource Name (ARN) de la clé AWS Key Management Service (AWS KMS key) que vous souhaitez utiliser.

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

Pour de plus amples informations sur l'utilisation d'une clé CMK pour chiffrer les données de session, consultez Activer le chiffrement des données de session par clé KMS (console).

Si vous ne voulez pas utiliser le chiffrement AWS KMS de vos données de session, vous pouvez supprimer le contenu suivant à partir de la politique :

```
,
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "key-name"
        }
```

8. Choisissez Next: Tags (Suivant : Balises).
9. (Facultatif) Ajoutez des identifications en choisissant Add tag (Ajouter une identification), et en saisissant les identifications préférées de la politique.
10. Choisissez Next: Review (Suivant : Vérification).
11. Sur la page Examiner une politique, dans le champ Nom, saisissez un nom pour la politique en ligne, tel que **`SessionManagerPermissions`**.
12. (Facultatif) Dans le champ Description, saisissez une description pour la politique.

Sélectionnez Créer une politique.

Pour de plus amples informations sur les actions `ssmmessages`, consultez Référence : ec2messages, ssmmessages et autres opérations d'API.

Création d'un rôle IAM personnalisé pour Session Manager

Vous pouvez créer un rôle AWS Identity and Access Management (IAM) qui accorde Session Manager l'autorisation d'effectuer des actions sur vos instances gérées Amazon EC2. Vous pouvez également inclure une politique pour accorder les autorisations nécessaires pour que les journaux de session soient envoyés à Amazon Simple Storage Service (Amazon S3) et Amazon CloudWatch Logs.

Après avoir créé le rôle IAM, pour plus d'informations sur la façon d'attacher le rôle à une instance, voir Attacher ou remplacer un profil d'instance sur le AWS re:Post site Web. Pour plus d'informations sur les profils d'instance et les rôles IAM, veuillez consulter les rubriques Utilisation de profils d'instance dans le Guide de l'utilisateur IAM et Rôles IAM pour Amazon EC2 dans le Guide de l'utilisateur Amazon Elastic Compute Cloud pour les instances Linux. Pour plus d'informations sur la création d'un rôle de service IAM pour les machines sur site, voir Créer le rôle de service IAM requis pour Systems Manager dans les environnements hybrides et multicloud.

Rubriques

- Création d'un rôle IAM avec les autorisations Session Manager minimales (console)
- Création d'un rôle IAM avec des autorisations pour Session Manager Amazon S3 et CloudWatch Logs (console)

Création d'un rôle IAM avec les autorisations Session Manager minimales (console)

Procédez comme suit pour créer un rôle IAM personnalisé avec une politique qui autorise uniquement des actions Session Manager sur vos instances.

Création d'un profil d'instance avec des autorisations Session Manager minimales

1. Connectez-vous à la console IAM AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le volet de navigation, sélectionnez Politiques, puis Créer une politique. (Si un bouton Get Started [Mise en route] est affiché, sélectionnez-le, puis Create Policy [Créer une politique].)

3. Sélectionnez l'onglet JSON.
4. Remplacez le contenu par défaut par la politique suivante. Pour chiffrer les données de session à l'aide de AWS Key Management Service (AWS KMS), remplacez *key-name* par le Amazon Resource Name (ARN) de la ressource AWS KMS key que vous souhaitez utiliser.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:UpdateInstanceInformation",
                "ssmmessages:CreateControlChannel",
                "ssmmessages:CreateDataChannel",
                "ssmmessages:OpenControlChannel",
                "ssmmessages:OpenDataChannel"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "key-name"
        }
    ]
}
```

Pour de plus amples informations sur l'utilisation d'une clé CMK pour chiffrer les données de session, consultez Activer le chiffrement des données de session par clé KMS (console).

Si vous n'avez pas l'intention d'utiliser le AWS KMS chiffrement pour les données de votre session, vous pouvez supprimer le contenu suivant de la politique.

```
,
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
```

```
            "Resource": "key-name"
        }
```

5. Choisissez Suivant : Balises.
6. (Facultatif) Ajoutez des identifications en choisissant Add tag (Ajouter une identification), et en saisissant les identifications préférées de la politique.
7. Choisissez Suivant : vérification.
8. Sur la page Examiner une politique, dans le champ Nom, saisissez un nom pour la politique en ligne, tel que **`SessionManagerPermissions`**.
9. (Facultatif) Dans le champ Description, saisissez une description pour la politique.
10. Sélectionnez Créer une politique.
11. Dans le volet de navigation, sélectionnez Rôles, puis Créer un rôle.
12. Sur la page Créer un rôle, choisissez Service AWS , et pour Cas d'utilisation, choisissez EC2.
13. Choisissez Suivant.
14. Sur la page Attached permissions policy (Politique d'autorisations attachée), cochez la case située à gauche du nom de la politique que vous venez de créer, tel que **`SessionManagerPermissions`**.
15. Choisissez Suivant.
16. Sur la page Review (Vérifier), pour Role name (Nom du rôle), saisissez un nom pour le rôle IAM, tel que **`MySessionManagerRole`**.
17. (Facultatif) Dans le champ Description du rôle, saisissez une description pour le profil d'instance.
18. (Facultatif) Ajoutez des identifications en choisissant Add tag (Ajouter une identification), et en saisissant les identifications préférées du rôle.

    Sélectionnez Créer un rôle.

Pour de plus amples informations sur les actions `ssmmessages`, veuillez consulter Référence : ec2messages, ssmmessages et autres opérations d'API.

Création d'un rôle IAM avec des autorisations pour Session Manager Amazon S3 et CloudWatch Logs (console)

Procédez comme suit pour créer un rôle IAM personnalisé avec une politique qui autorise des actions Session Manager sur vos instances. La politique 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 de journaux CloudWatch Amazon Logs.

> **Important**
>
> Pour produire les journaux de session dans un compartiment Amazon S3 appartenant à un autre Compte AWS, vous devez ajouter l'autorisation `s3:PutObjectAcl` à la politique de rôle IAM. En outre, vous devez vous assurer que la politique relative aux compartiments accorde un accès intercompte au rôle IAM utilisé par le compte propriétaire pour accorder des autorisations Systems Manager aux instances gérées. Si le compartiment utilise le chiffrement KMS (Key Management Service), la politique KMS du compartiment doit également accorder cet accès intercompte. Pour plus d'informations sur la configuration des autorisations de compartiment intercomptes dans Amazon S3, veuillez consulter la rubrique Accorder des autorisations intercomptes sur un compartiment du Guide de l'utilisateur Amazon Simple Storage Service. Si les autorisations intercomptes ne sont pas ajoutées, le compte qui possède le compartiment Amazon S3 ne peut pas accéder aux journaux de sortie de la session.

Pour en savoir plus sur la spécification des préférences de stockage des journaux de session, consultez Activation et désactivation de la journalisation des activités de session.

Pour créer un rôle IAM avec des autorisations pour Session Manager Amazon S3 et CloudWatch Logs (console)

1. Connectez-vous à la console IAM AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le volet de navigation, sélectionnez Politiques, puis Créer une politique. (Si un bouton Get Started [Mise en route] est affiché, sélectionnez-le, puis Create Policy [Créer une politique].)
3. Sélectionnez l'onglet JSON.
4. Remplacez le contenu par défaut par la politique suivante. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

```
{
    "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-prefix/*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetEncryptionConfiguration"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "key-name"
        },
        {
            "Effect": "Allow",
            "Action": "kms:GenerateDataKey",
            "Resource": "*"
```

```
        }
    ]
}
```

5. Choisissez Suivant : Balises.
6. (Facultatif) Ajoutez des identifications en choisissant Add tag (Ajouter une identification), et en saisissant les identifications préférées de la politique.
7. Choisissez Suivant : vérification.
8. Sur la page Examiner une politique, dans le champ Nom, saisissez un nom pour la politique en ligne, tel que **`SessionManagerPermissions`**.
9. (Facultatif) Dans le champ Description, saisissez une description pour la politique.
10. Sélectionnez Créer une politique.
11. Dans le volet de navigation, sélectionnez Rôles, puis Créer un rôle.
12. Sur la page Créer un rôle, choisissez Service AWS , et pour Cas d'utilisation, choisissez EC2.
13. Choisissez Suivant.
14. Sur la page Attached permissions policy (Politique d'autorisations attachée), cochez la case située à gauche du nom de la politique que vous venez de créer, tel que **`SessionManagerPermissions`**.
15. Choisissez Suivant.
16. Sur la page Review (Vérifier), pour Role name (Nom du rôle), saisissez un nom pour le rôle IAM, tel que **`MySessionManagerRole`**.
17. (Facultatif) Dans le champ Role description (Description du rôle), saisissez la description du nouveau rôle.
18. (Facultatif) Ajoutez des identifications en choisissant Add tag (Ajouter une identification), et en saisissant les identifications préférées du rôle.
19. Sélectionnez Créer un rôle.

Étape 3 : Contrôler les accès de session aux nœuds gérés

Vous accordez ou révoquez l'accès de Session Manager aux nœuds gérés en utilisant des politiques AWS Identity and Access Management (IAM). Vous pouvez créer une politique et l'associer à un utilisateur ou à un groupe IAM qui spécifie les nœuds gérés auxquels l'utilisateur ou le groupe peut se connecter. Vous pouvez également spécifier les opérations d'API Session Manager que l'utilisateur ou les groupes peuvent effectuer sur ces nœuds gérés.

Pour vous aider à démarrer avec les politiques d'autorisations IAM pour Session Manager, nous avons créé des exemples de politiques pour un utilisateur final et un utilisateur administrateur. Vous ne pouvez utiliser ces politiques qu'avec des modifications mineures. Vous pouvez également les utiliser comme guide pour créer des politiques IAM personnalisées. Pour plus d'informations, consultez Exemple de stratégies IAM pour Session Manager. Pour obtenir des informations sur la création de politiques IAM et la façon de les attacher à des utilisateurs ou des groupes, consultez Création de politiques IAM et Ajout et suppression de politiques IAM dans le Guide de l'utilisateur IAM.

À propos des formats ARN des identifiants de session

Lorsque vous créez une politique IAM pour l'accès de Session Manager, vous spécifiez un ID de session dans le cadre d'Amazon Resource Name (ARN). L'ID de session inclut le nom d'utilisateur en tant que variable. Pour illustrer cela, voici le format d'un ARN Session Manager et un exemple :

```
arn:aws:ssm:region-id:account-id:session/session-id
```

Par exemple :

```
arn:aws:ssm:us-east-2:123456789012:session/JohnDoe-1a2b3c4d5eEXAMPLE
```

Pour de plus amples informations sur l'utilisation de variables dans les politiques IAM, consultez Éléments de politique IAM : Variables.

Rubriques

- Démarrer une session shell par défaut en spécifiant le document de session par défaut dans les politiques IAM
- Démarrer une session avec un document en spécifiant les documents de session dans les politiques IAM
- Exemple de stratégies IAM pour Session Manager
- Exemples de politiques IAM supplémentaires pour Session Manager

Démarrer une session shell par défaut en spécifiant le document de session par défaut dans les politiques IAM

Lorsque vous configurez Session Manager pour votre session Compte AWS ou lorsque vous modifiez les préférences de session dans la console Systems Manager, le système crée un

document de session SSM appelé`SSM-SessionManagerRunShell`. Il s'agit du document de session par défaut. Session Manager utilise ce document pour enregistrer vos préférences de session, qui incluent des informations telles que les suivantes :

- Emplacement dans lequel vous souhaitez enregistrer les données de session, tel qu'un bucket Amazon Simple Storage Service (Amazon S3) ou un groupe de journaux CloudWatch Amazon Logs.
- Un identifiant de clé AWS Key Management Service (AWS KMS) pour chiffrer les données de session.
- Si la prise en charge Run As est autorisée pour vos sessions.

Voici un exemple des informations contenues dans le document des préférences de session `SSM-SessionManagerRunShell`.

```
{
  "schemaVersion": "1.0",
  "description": "Document to hold regional settings for Session Manager",
  "sessionType": "Standard_Stream",
  "inputs": {
    "s3BucketName": "DOC-EXAMPLE-BUCKET",
    "s3KeyPrefix": "MyS3Prefix",
    "s3EncryptionEnabled": true,
    "cloudWatchLogGroupName": "MyCWLogGroup",
    "cloudWatchEncryptionEnabled": false,
    "kmsKeyId": "1a2b3c4d",
    "runAsEnabled": true,
    "runAsDefaultUser": "RunAsUser"
  }
}
```

Par défaut, Session Manager utilise le document de session par défaut lorsqu'un utilisateur démarre une session à partir de la AWS Management Console. Cela s'applique Fleet Manager soit Session Manager à la console Systems Manager, soit à EC2 Connect dans la console Amazon EC2. Session Managerutilise également le document de session par défaut lorsqu'un utilisateur démarre une session à l'aide d'une AWS CLI commande comme dans l'exemple suivant :

```
aws ssm start-session \
    --target i-02573cafcfEXAMPLE
```

Pour démarrer une session shell par défaut, vous devez spécifier le document de session par défaut dans la politique IAM, comme illustré dans l'exemple suivant.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "EnableSSMSession",
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ec2:us-west-2:123456789012:instance/i-02573cafcfEXAMPLE",
                "arn:aws:ssm:us-west-2:123456789012:document/SSM-
SessionManagerRunShell"
            ]
        }
    ]
}
```

Démarrer une session avec un document en spécifiant les documents de session dans les politiques IAM

Si vous utilisez la commande AWS CLI start-session en utilisant le document de session par défaut, vous pouvez omettre le nom du document. Le système appelle automatiquement le document de session `SSM-SessionManagerRunShell`.

Dans tous les autres cas, vous devez spécifier une valeur pour le paramètre `document-name`. Lorsqu'un utilisateur indique le nom d'un document de session dans une commande, le système vérifie sa politique IAM pour vérifier qu'il est autorisé à accéder au document. S'il n'est pas autorisé, la demande de connexion échoue. Les exemples suivants incluent le paramètre `document-name` dans le document de session `AWS-StartPortForwardingSession`.

```
aws ssm start-session \
    --target i-02573cafcfEXAMPLE \
    --document-name AWS-StartPortForwardingSession \
    --parameters '{"portNumber":["80"], "localPortNumber":["56789"]}'
```

Appliquer une vérification des autorisations de document de session lors du démarrage d'une session

Pour restreindre l'accès au document de session `AWS-StartPortForwardingSession`, 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 `start-session`. L'élément de condition suivant, lorsqu'il est ajouté à l'action `ssm:StartSession` dans la politique IAM, procède à une vérification de l'accès au document de session.

```
"Condition": {
    "BoolIfExists": {
        "ssm:SessionDocumentAccessCheck": "true"
    }
}
```

Si cet élément de condition est défini sur `true`, l'accès explicite à un document de session doit être accordé dans la stratégie IAM pour que l'utilisateur puisse démarrer une session. Pour garantir l'application de l'élément de condition, il est impératif de l'inclure dans toutes les déclarations de politique qui autorisent l'action `ssm:StartSession`. Voici un exemple.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "EnableSSMSession",
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ec2:us-west-2:123456789012:instance/i-02573cafcfEXAMPLE",
                "arn:aws:ssm:us-west-2::document/AWS-StartPortForwardingSession"
            ],
            "Condition": {
                "BoolIfExists": {
                    "ssm:SessionDocumentAccessCheck": "true"
                }
            }
        }
    ]
}
```

Avec cette politique IAM en place, si l'élément de condition `SessionDocumentAccessCheck` est défini sur `true`, les utilisateurs doivent entrer le paramètre `document-name` dans leur commande lorsqu'ils démarrent une session avec AWS CLI. La valeur de `document-name` doit être le document spécifié dans la section `Resource` de la politique IAM. Si l'utilisateur saisit un autre nom de document ou s'il ne spécifie pas le paramètre `document-name`, la demande échoue.

Si l'élément de condition `SessionDocumentAccessCheck` est défini sur `false`, cela n'affecte pas l'évaluation de la politique IAM.

Pour obtenir un exemple de spécification d'un document de session Session Manager dans une politique IAM, consultez Démarrage rapide - Politiques d'utilisateur final pour Session Manager.

Autres scénarios

Pour démarrer une session à l'aide de SSH, les étapes de configuration doivent être effectuées sur le nœud géré cible et sur la machine locale de l'utilisateur. Pour plus d'informations, voir (facultatif) Autoriser et contrôler les autorisations pour les connexions SSH via Session Manager.

Exemple de stratégies IAM pour Session Manager

Utilisez les exemples de cette section pour vous aider à créer des politiques AWS Identity and Access Management (IAM) fournissant les autorisations d'Session Manageraccès les plus couramment requises.

> **Note**
>
> Vous pouvez également utiliser une AWS KMS key politique pour contrôler les entités IAM (utilisateurs ou rôles) qui Comptes AWS ont accès à votre clé KMS. Pour plus d'informations, consultez la section Présentation de la gestion de l'accès à vos AWS KMS ressources et de l'utilisation des politiques clés AWS KMS dans le guide du AWS Key Management Service développeur.

Rubriques

- Démarrage rapide - Politiques d'utilisateur final pour Session Manager
- Démarrage rapide - Politique d'administrateur pour Session Manager

Démarrage rapide - Politiques d'utilisateur final pour Session Manager

Utilisez les exemples suivants pour créer des politiques d'utilisateur final IAM pour Session Manager.

Vous pouvez créer une politique qui permet aux utilisateurs de démarrer des sessions uniquement à partir de la Session Manager console et AWS Command Line Interface (AWS CLI), uniquement à partir de la console Amazon Elastic Compute Cloud (Amazon EC2), ou à partir des trois.

Ces politiques permettent aux utilisateurs finaux de démarrer une session sur un nœud géré particulier et de mettre fin à leurs propres sessions uniquement. Consultez la page Exemples de politiques IAM supplémentaires pour Session Manager pour obtenir des exemples de personnalisation de la politique.

Dans les exemples de stratégies suivants, remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

Reportez-vous aux sections suivantes pour voir des exemples de politiques pour la plage d'accès aux sessions que vous souhaitez fournir.

Gestionnaire de session and Fleet Manager

Utilisez cet exemple de politique pour permettre aux utilisateurs de démarrer et de reprendre des sessions uniquement à partir des Fleet Manager consoles Session Manager et.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ec2:region:account-id: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",
                "ssm:ResumeSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:userid}-*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [

 "kms:GenerateDataKey" (3)
            ],
            "Resource": "key-name"
        }
    ]
}
```

Amazon EC2

Utilisez cet exemple de politique pour permettre aux utilisateurs de démarrer des sessions uniquement à partir de la console Amazon EC2. Cette politique 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:region:account-id:instance/instance-id",
                "arn:aws:ssm:region:account-id:document/SSM-
SessionManagerRunShell" 1
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetConnectionStatus",
                "ssm:DescribeInstanceInformation"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession",
                "ssm:ResumeSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:userid}-*"
            ]
        }
    ]
}
```

AWS CLI

Utilisez cet exemple de politique pour permettre aux utilisateurs de démarrer et de reprendre des sessions à partir du AWS CLI.

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

```
 "ssm:SendCommand" 4
            ],
            "Resource": [
                "arn:aws:ec2:region:account-id:instance/instance-id",
                "arn:aws:ssm:region:account-id:document/SSM-
SessionManagerRunShell" 1
            ],
            "Condition": {
                "BoolIfExists": {
                    "ssm:SessionDocumentAccessCheck":
 "true" 2
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession",
                "ssm:ResumeSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:userid}-*"
            ]
        },
        {
            "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 politique à la place. Vous pouvez également spécifier le document AWS fourni aux utilisateurs qui `AWS-StartSSHSession` démarrent des sessions à l'aide de SSH. Pour plus d'informations sur les étapes de configuration nécessaires pour

prendre en charge les sessions utilisant SSH, voir (facultatif) Autoriser et contrôler les autorisations pour les connexions SSH via le protocole SSH. Session Manager

[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 Appliquer une vérification des autorisations de document de session lors du démarrage d'une session.

[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 comptez utiliser le chiffrement AWS Key Management Service (AWS KMS) pour les données de votre session, remplacez *key-name* par le Amazon Resource Name (ARN) de la clé KMS que vous souhaitez utiliser, au format indiqué. `arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-12345EXAMPLE` Si vous ne voulez pas utiliser le chiffrement de clé KMS pour vos données de session, supprimez le contenu suivant de la politique.

```
,
        {
            "Effect": "Allow",
            "Action": [
                "kms:GenerateDataKey"
            ],
            "Resource": "key-name"
        }
```

Pour plus d'informations sur l'utilisation AWS KMS pour le chiffrement des données de session, consultezActiver le chiffrement des données de session par clé KMS (console).

[4] L'autorisation pour SendCommandest nécessaire dans les cas où un utilisateur tente de démarrer une session depuis la console Amazon EC2, mais elle SSM Agent doit être mise à jour vers la version minimale requise pour Session Manager la première fois. Run Commandest utilisé pour envoyer une commande à l'instance afin de mettre à jour l'agent.

Démarrage rapide - Politique d'administrateur pour Session Manager

Utilisez les exemples suivants pour créer des politiques d'administrateur IAM pour Session Manager.

Ces politiques autorisent les administrateurs à démarrer une session sur les nœuds gérés qui sont balisées avec `Key=Finance,Value=WebServers`, à créer, mettre à jour et supprimer des

préférences, et à mettre fin à leurs propres sessions uniquement. Consultez la page Exemples de politiques IAM supplémentaires pour Session Manager pour obtenir des exemples de personnalisation de la politique.

Vous pouvez créer une politique qui permet aux administrateurs d'effectuer ces tâches uniquement à partir de la Session Manager console et AWS CLI uniquement à partir de la console Amazon EC2, ou à partir des trois.

Dans les exemples de stratégies suivants, remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

Reportez-vous aux sections suivantes pour voir des exemples de politiques pour les trois scénarios d'autorisations.

Gestionnaire de session and CLI

Utilisez cet exemple de politique 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 politique 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:region:account-id: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",
                "ssm:StartSession"
            ],
            "Resource": "arn:aws:ssm:region:account-id:document/SSM-
SessionManagerRunShell"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession",
                "ssm:ResumeSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:userid}-*"
            ]
        }
    ]
}
```

Amazon EC2

Utilisez cet exemple de politique pour permettre aux administrateurs d'effectuer les tâches liées à la session uniquement à partir de la console Amazon EC2. Cette politique 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" 1
            ],
            "Resource": [
                "arn:aws:ec2:region:account-id:instance/*"
            ],
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/tag-key": [
                        "tag-value"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ssm:region:account-id:document/SSM-SessionManagerRunShell"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetConnectionStatus",
                "ssm:DescribeInstanceInformation"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession",
                "ssm:ResumeSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:userid}-*"
            ]
        }
```

```
    ]
}
```

Gestionnaire de session, CLI, and Amazon EC2

Utilisez cet exemple de politique 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" (1)
            ],
            "Resource": [
                "arn:aws:ec2:region:account-id: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",
                "ssm:StartSession"
            ],
            "Resource": "arn:aws:ssm:region:account-id:document/SSM-
SessionManagerRunShell"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession",
                "ssm:ResumeSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:userid}-*"
            ]
        }
    ]
}
```

[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'SSM Agent.

Exemples de politiques IAM supplémentaires pour Session Manager

Les exemples de politiques suivants vous accompagnent dans la création d'une politique AWS Identity and Access Management (IAM) personnalisée pour tous les scénarios d'accès utilisateur à Session Manager que vous souhaitez prendre en charge.

Rubriques

- Exemple 1 : octroi d'un accès aux documents depuis la console
- Exemple 2 : restriction de l'accès à des nœuds gérés spécifiques
- Exemple 3 : restriction de l'accès en fonction des balises
- Exemple 4 : autorisation d'un utilisateur à mettre fin uniquement aux sessions qu'il a démarrées
- Exemple 5 : octroi d'un accès complet (administrateur) à toutes les sessions

Exemple 1 : octroi d'un accès aux documents depuis la console

Vous pouvez autoriser les utilisateurs à spécifier un document personnalisé lorsqu'ils lancent une session à l'aide de la console Session Manager. L'exemple de politique IAM suivant accorde l'autorisation d'accéder à des documents dont le nom commence par **`SessionDocument-`** dans la Région AWS et sur le Compte AWS spécifiés.

Pour utiliser cette politique, remplacez chaque *`exemple d'espace réservé pour les ressources`* par vos propres informations.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetDocument",
                "ssm:ListDocuments"
            ],
            "Resource": [
                "arn:aws:ssm:region:account-id:document/SessionDocument-*"

            ],
            "Condition": {
                "BoolIfExists": {
                    "ssm:SessionDocumentAccessCheck": "true"
                }
            }
        }
    ]
}
```

> **Note**
>
> La console Session Manager ne prend en charge que les documents de session dont le `sessionType` et `Standard_Stream` et utilisés pour définir les préférences de session. Pour plus d'informations, consultez Schéma de document de session.

Exemple 2 : restriction de l'accès à des nœuds gérés spécifiques

Vous pouvez créer une politique IAM qui définit les nœuds gérés auxquels un utilisateur est autorisé à se connecter à l'aide du gestionnaire de session. Par exemple, la politique suivante accorde à un utilisateur l'autorisation de démarrer, de terminer et de reprendre ses sessions sur trois nœuds spécifiques. La politique interdit à l'utilisateur de se connecter à des nœuds autres que ceux spécifiés.

> **Note**
>
> Pour les utilisateurs fédérés, voirExemple 4 : autorisation d'un utilisateur à mettre fin uniquement aux sessions qu'il a démarrées.

```
{
    "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",
                "arn:aws:ssm:us-east-2:123456789012:document/SSM-
SessionManagerRunShell"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession",
                "ssm:ResumeSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:userid}-*"
            ]
        }
    ]
```

```
}
```

Exemple 3 : restriction de l'accès en fonction des balises

Vous pouvez restreindre l'accès à des nœuds gérés en fonction de balises spécifiques. Dans l'exemple suivant, l'utilisateur est autorisé à démarrer et à reprendre des sessions (`Effect: Allow, Action: ssm:StartSession, ssm:ResumeSession`) sur n'importe quel nœud géré (`Resource: arn:aws:ec2:`*`region`*`:`*`987654321098`*`:instance/*`) à condition que le nœud soit un nœud Finance WebServer (`ssm:resourceTag/Finance: WebServer`). Si l'utilisateur envoie une commande à un nœud géré non balisé 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:us-east-2:123456789012:instance/*"
            ],
            "Condition": {
                "StringLike": {
                    "ssm:resourceTag/Finance": [
                        "WebServers"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:TerminateSession",
                "ssm:ResumeSession"
            ],
            "Resource": [
                "arn:aws:ssm:*:*:session/${aws:userid}-*"
            ]
        },
        {
```

```
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
            "Resource": [
                "arn:aws:ssm:us-east-2:123456789012:document/SSM-
SessionManagerRunShell"
            ]
        }
    ]
}
```

Vous pouvez créer des politiques IAM qui permettent à un utilisateur de démarrer des sessions sur des nœuds gérés qui contiennent plusieurs balises. La politique suivante permet à l'utilisateur de démarrer des sessions sur des nœuds gérés qui contiennent les balises spécifiées. Si un utilisateur envoie une commande à un nœud géré 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"
               ]
            }
         }
      },
      {
            "Effect": "Allow",
            "Action": [
                "ssm:StartSession"
            ],
```

```
            "Resource": [
                "arn:aws:ssm:us-east-2:123456789012:document/SSM-
SessionManagerRunShell"
            ]
        }
    ]
}
```

Pour plus d'informations sur la création de politiques IAM, consultez Politiques gérées et politiques en ligne dans le Guide de l'utilisateur IAM. Pour plus d'informations sur le balisage des nœuds gérés, consultez Balisage des nœuds gérés la section Marquage de vos ressources Amazon EC2 dans le guide de l'utilisateur Amazon EC2 (le contenu s'applique aux Windows nœuds gérés et aux nœuds gérés). Linux Pour plus d'informations sur le renforcement de votre posture de sécurité vis-à-vis des commandes de niveau racine non autorisées sur vos nœuds gérés, consultez Limitation de l'accès aux commandes de niveau racine via l'SSM Agent.

Exemple 4 : autorisation d'un utilisateur à mettre fin uniquement aux sessions qu'il a démarrées

Session Managerpropose deux méthodes pour contrôler les sessions auxquelles un utilisateur fédéré de votre entreprise Compte AWS est autorisé à mettre fin.

- Utilisez la variable `{aws:userid}` dans une politique d'autorisation AWS Identity and Access Management (IAM). Les utilisateurs fédérés ne peuvent terminer que les sessions qu'ils ont démarrées. Pour les utilisateurs non fédérés, utilisez la variable `{aws:username}` au lieu de `{aws:userid}`.
- Utilisez les balises fournies par les AWS balises dans une politique d'autorisation IAM. Dans la politique, vous incluez une condition qui permet aux utilisateurs de ne terminer que les sessions marquées avec des balises spécifiques fournies par AWS. Cette méthode fonctionne pour tous les comptes, y compris ceux qui utilisent des ID fédérés pour accorder l'accès à AWS.

Méthode 1 : octroyer TerminateSession des privilèges à l'aide de la variable **`{aws:username}`**

La politique IAM suivante permet à un utilisateur d'afficher les ID de toutes les sessions de votre compte. Toutefois, les utilisateurs ne peuvent interagir avec les nœuds gérés que par le biais des sessions qu'ils ont démarrées. Un utilisateur auquel vous attribuez la politique suivante ne peut pas se connecter ni mettre fin aux sessions des autres utilisateurs. Cette politique utilise la variable `{aws:username}`.

> **Note**
>
> Cette méthode ne fonctionne pas pour les comptes qui accordent l'accès à AWS à l'aide d'ID fédérés.

```
{
    "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 : octroyer TerminateSession des privilèges à l'aide de balises fournies par AWS

Vous pouvez contrôler les sessions qu'un utilisateur peut terminer en utilisant des variables de clé de balise conditionnelle spécifiques dans une politique utilisateur IAM. 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 de vos utilisateurs Compte AWS 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 politique de contrôle d'accès IAM. Vous devez utiliser les balises de ressources fournies par AWS, décrites ci-dessous.

`aws:ssmmessages:target-id`

Avec cette clé de balise, vous incluez l'ID de nœud géré comme valeur dans la politique. Dans le bloc de politique suivant, l'instruction de condition permet à un utilisateur de mettre fin uniquement au nœud 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 politique 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 de votre Compte AWS. Pour récupérer l'identifiant d'un utilisateur dans votre Compte AWS, utilisez la commande IAM,`get-user`. Pour plus d'informations, voir get-user dans la AWS Identity and Access Management section du guide de l'utilisateur IAM.

```
{
     "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 politique pour les cas où le type d'appelant est `AssumedRole`. Vous pouvez utiliser la variable `{aws:userid}` pour la valeur que vous attribuez à `aws:ssmmessages:session-id`. Vous pouvez également coder en dur un ID de rôle pour la valeur que vous attribuez à `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 ne peut contenir que les caractères suivants : Lettres Unicode, 0-9 _, espace ., :, /, =, +, -, @ et \.

Pour récupérer l'ID de rôle d'un rôle dans votre Compte AWS, utilisez la `get-caller-identity` commande. Pour plus d'informations, consultez get-caller-identity dans la référence des commandes. AWS CLI

```
{
     "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 des politiques IAM qui permettent à un utilisateur de terminer des sessions marquées 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":[
                  "i-02573cafcfEXAMPLE"
               ],
               "ssm:resourceTag/aws:ssmmessages:session-id":[
                  "${aws:userid}*"
               ]
            }
         }
      }
   ]
}
```

Exemple 5 : octroi d'un accès complet (administrateur) à toutes les sessions

La politique IAM suivante permet à un utilisateur d'interagir pleinement avec tous les nœuds gérés et toutes les sessions créées par l'ensemble des utilisateurs des nœuds. 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 : Configuration des préférences de session

Les utilisateurs auxquels des autorisations administratives ont été accordées dans leur politique AWS Identity and Access Management (IAM) peuvent configurer les préférences de session, notamment les suivantes :

- Activation d'Exécuter en tant que support pour les nœuds gérés Linux. Cela permet de démarrer des sessions en utilisant les informations d'identification d'un utilisateur du système d'exploitation spécifié au lieu des informations d'identification d'un `ssm-user` compte généré par le système qui AWS Systems Manager Session Manager peut être créé sur un nœud géré.
- Configurez Session Manager pour utiliser AWS KMS key le chiffrement afin de fournir une protection supplémentaire aux données transmises entre les machines clientes et les nœuds gérés.
- Configurez Session Manager pour créer et envoyer des journaux d'historique de session vers un bucket Amazon Simple Storage Service (Amazon S3) ou un groupe de journaux CloudWatch Amazon 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 nœuds gérés et les commandes exécutées au cours des sessions.
- Configurer les délais d'expiration de session. Vous pouvez utiliser ce paramètre pour spécifier quand mettre fin à une session après une période d'inactivité.
- Configurez Session Manager de sorte à utiliser des profils de shell configurables. Ces profils personnalisables vous permettent de définir des préférences dans les sessions telles que les préférences du shell, les variables d'environnement, les répertoires de travail et l'exécution de plusieurs commandes au démarrage d'une session.

Pour plus d'informations sur les autorisations qui sont nécessaires pour configurer les préférences Session Manager, veuillez consulter la rubrique the section called “Accorder ou révoquer des autorisations utilisateur pour mettre à jour des préférences Session Manager”.

Rubriques

- Accorder ou révoquer des autorisations utilisateur pour mettre à jour des préférences Session Manager

- Spécifier une valeur de délai d'expiration d'une session inactive
- Spécification de la durée de session maximale
- Autoriser les profils de shell configurables
- Activez Exécuter en tant que support pour Linux les nœuds macOS gérés
- Activer le chiffrement des données de session par clé KMS (console)
- Création d'un document de préférences Session Manager (ligne de commande)
- Mettre à jour les préférences Session Manager (ligne de commande)

Pour plus d'informations sur l'utilisation de la console Systems Manager pour configurer des options pour la journalisation des données de session, consultez les rubriques suivantes.

- Journalisation des données de session avec Amazon S3 (console)
- Données de session de streaming à l'aide d'Amazon CloudWatch Logs (console)
- Enregistrement des données de session à l'aide d'Amazon CloudWatch Logs (console)

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 AWS Systems Manager (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. Les autorisations sont accordées via une politique AWS Identity and Access Management (IAM).

Politique administrateur pour autoriser la création et la mise à jour des préférences

Un administrateur peut utiliser la politique suivante pour créer et mettre à jour des préférences à tout moment. La politique 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:GetDocument",
```

```
                "ssm:UpdateDocument",
                "ssm:DeleteDocument"
            ],
            "Effect": "Allow",
            "Resource": [
                "arn:aws:ssm:us-east-2:123456789012:document/SSM-
SessionManagerRunShell"
            ]
        }
    ]
}
```

Politique utilisateur pour interdire la mise à jour des préférences

Utilisez la politique 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:GetDocument",
                "ssm:UpdateDocument",
                "ssm:DeleteDocument"
            ],
            "Effect": "Deny",
            "Resource": [
                "arn:aws:ssm:us-east-2:123456789012:document/SSM-
SessionManagerRunShell"
            ]
        }
    ]
}
```

Spécifier une valeur de délai d'expiration d'une session inactive

Session Manager, une fonctionnalité de AWS Systems Manager, vous permet de spécifier le temps nécessaire à un utilisateur pour passer à l'état inactif avant que le système mette fin à une session. Par défaut, les sessions expirent au bout de 20 minutes d'inactivité. Vous pouvez modifier ce paramètre de sorte à spécifier qu'une session expire entre 1 et 60 minutes d'inactivité. Certaines

agences de sécurité informatique professionnelles recommandent de définir des délais d'inactivité de session de 15 minutes maximum.

Pour autoriser le délai d'expiration d'une session inactive (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Session Manager.
3. Sélectionnez l'onglet Préférences, puis Modifier.
4. Spécifiez le temps nécessaire à un utilisateur pour passer à l'état inactif avant qu'une session se termine, dans le champ minutes, sous Délai d'expiration d'une session inactive.
5. Choisissez Enregistrer.

Spécification de la durée de session maximale

Session Manager, une fonctionnalité de AWS Systems Manager, vous permet de spécifier la durée maximale d'une session avant qu'elle ne se termine. Par défaut, les sessions n'ont pas de durée maximale. La valeur que vous spécifiez pour la durée maximale de session doit être comprise entre 1 et 1 440 minutes.

Pour spécifier la durée de session maximale (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Session Manager.
3. Sélectionnez l'onglet Préférences, puis Modifier.
4. Activez la case à cocher en regard de Enable maximum session duration (Activer la durée de session maximale).
5. Spécifiez la durée maximale de la session avant son terme dans le champ minutes sous Maximum session duration (Durée de session maximale).
6. Sélectionnez Enregistrer.

Autoriser les profils de shell configurables

Par défaut, les sessions sur les instances EC2 pour Linux commencent à utiliser le shell Bourne (sh). Vous préférerez peut-être utiliser un autre shell, bash par exemple. En autorisant les profils

de shell configurables, vous pouvez personnaliser des préférences de session, telles que des préférences de shell, des variables d'environnement, des répertoires de travail et l'exécution de plusieurs commandes au démarrage d'une session.

> **Important**
>
> Systems Manager ne vérifie pas les commandes ou scripts de votre profil shell pour voir les modifications qu'ils apporteraient à une instance avant leur exécution. Pour limiter la capacité d'un utilisateur à modifier des commandes ou scripts entrés dans son profil shell, nous vous recommandons de procéder comme suit :
>
> - Créez un document de type session personnalisé pour vos utilisateurs et rôles AWS Identity and Access Management (IAM). Modifiez ensuite la politique IAM pour ces utilisateurs et rôles de sorte que l'opération d'API `StartSession` puisse seulement utiliser le document de type session que vous avez créé pour ceux-ci. Pour plus d'informations, consultez Création d'un document de préférences Session Manager (ligne de commande) et Démarrage rapide - Politiques d'utilisateur final pour Session Manager.
> - Modifiez la politique IAM pour vos utilisateurs et rôles IAM afin de refuser l'accès à l'opération d'API `UpdateDocument` pour la ressource de document de type session que vous créez. Vos utilisateurs et rôles sont ainsi autorisés à utiliser le document que vous avez créé pour leurs préférences de session mais pas à en modifier les paramètres.

Pour activer des profils de shell configurables

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Session Manager.
3. Sélectionnez l'onglet Préférences, puis Modifier.
4. Spécifiez les variables d'environnement, les préférences de shell ou les commandes que vous voulez exécuter au démarrage de votre session dans les champs des systèmes d'exploitation applicables.
5. Sélectionnez Enregistrer.

Voici quelques exemples des commandes qui peuvent être ajoutées à votre profil de shell.

Sur les instances Linux, passez au shell bash et au répertoire /usr.

```
exec /bin/bash
cd /usr
```

Affichez un horodatage et un message de bienvenue au démarrage d'une session.

Linux & macOS

```
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."
```

Affichez l'activité dynamique du système au démarrage d'une session.

Linux & macOS

```
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 "-------  ------    -----      ----- -----   ------     -- -----------"}
```

Activez Exécuter en tant que support pour Linux les nœuds macOS gérés

Par défaut, Session Manager authentifie les connexions lancées à l'aide des informations d'identification d'un compte `ssm-user` généré par le système qui est créé sur un nœud géré. (Sur les machines Linux et macOS, le compte est ajouté à `/etc/sudoers/`.) Si vous le souhaitez, vous pouvez authentifier des sessions en utilisant les informations d'identification d'un compte utilisateur de système d'exploitation (SE). Dans ce cas, le Gestionnaire de session vérifie que le compte de système d'exploitation (SE) que vous avez spécifié existe sur le nœud avant de démarrer la session. Si vous tentez de lancer une session à l'aide d'un compte de système d'exploitation (SE) qui n'existe pas sur le nœud, la connexion échoue.

> **Note**
>
> Le gestionnaire de session ne prend pas en charge l'utilisation d'un compte utilisateur `root` de système d'exploitation pour authentifier les connexions. Pour les sessions authentifiées à l'aide d'un compte utilisateur de système d'exploitation, les politiques au niveau du système d'exploitation et de répertoire du nœud, telles que les restrictions de connexion ou les restrictions d'utilisation des ressources système, peuvent ne pas s'appliquer.

Comment ça marche

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 entité IAM (utilisateur ou rôle) a-t-elle été balisée avec `SSMSessionRunAs = os user account name` ?

   Si oui, le nom d'utilisateur de système d'exploitation (SE) existe-t-il sur le nœud géré ? Si c'est le cas, démarrer la session. Si ce n'est pas le cas, ne pas autoriser une session à démarrer.

   Si l'entité IAM n'a pas été balisée avec `SSMSessionRunAs = os user account name`, passez à l'étape 2.
2. Si l'entité IAM n'a pas été balisée`SSMSessionRunAs = os user account name`, un nom d'utilisateur du système d'exploitation a-t-il été spécifié dans les Session Manager préférences Compte AWS de ?

   Si oui, le nom d'utilisateur de système d'exploitation (SE) existe-t-il sur le nœud géré ? Si c'est le cas, démarrer la session. Si ce n'est pas le cas, ne pas autoriser une session à démarrer.

> **Note**
>
> Lorsque vous activez la prise en charge de l'exécution en tant que, cela empêche le Gestionnaire de session de démarrer des sessions à l'aide du compte `ssm-user` sur un nœud géré. Cela signifie que si Session Manager ne réussit pas la connexion à l'aide du compte utilisateur du système d'exploitation spécifié, il ne reviendra pas à la connexion à l'aide de la méthode par défaut.
> Si vous activez l'exécution en tant que sans spécifier de compte de système d'exploitation (SE) ni baliser une entité IAM et que vous n'avez pas spécifié de compte de système d'exploitation (SE) dans les préférences du Gestionnaire de session, les tentatives de connexion à la session échoueront.

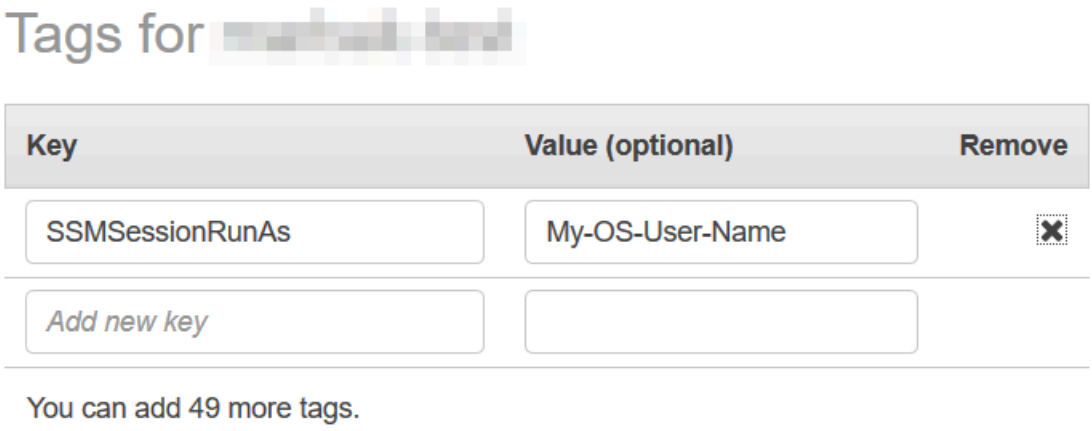
Pour activer Exécuter en tant que support pour les nœuds gérés Linux et macOS

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Session Manager.
3. Sélectionnez l'onglet Préférences, puis Modifier.
4. Cochez la case en regard de Activer Exécuter en tant que support des instances Linux.
5. Effectuez l'une des actions suivantes :

   - Option 1 : dans le champ Nom d'utilisateur de système d'exploitation, saisissez le nom du compte utilisateur de système d'exploitation (SE) que vous souhaitez utiliser pour démarrer les sessions. Avec cette option, toutes les sessions sont exécutées par le même utilisateur du système d'exploitation pour tous les utilisateurs de votre système Compte AWS qui se connectent en utilisantSession Manager.
   - Option 2 (Recommandé) : choisir le lien IAM console (Console IAM). Dans le panneau de navigation, sélectionnez Users (Utilisateurs) ou Roles (Rôles). Sélectionnez 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 de système d'exploitation (SE) pour la valeur de la clé. Sélectionnez Enregistrer les modifications.

     Avec cette option, vous pouvez spécifier des utilisateurs de système d'exploitation (SE) uniques pour différentes entités IAM si vous le souhaitez. Pour plus d'informations sur le balisage des entités IAM (utilisateurs ou rôles), consultez la section Balisage des ressources IAM dans le Guide de l'utilisateur IAM

Voici un exemple.



6. Choisissez Enregistrer.

Activer le chiffrement des données de session par clé KMS (console)

Utilisez AWS Key Management Service (AWS KMS) pour créer et gérer des clés de chiffrement. 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 nœuds gérés et les machines locales des utilisateurs de votre Compte AWS sont chiffrées à l'aide de la clé de chiffrement KMS. (Ceci s'ajoute au chiffrement TLS 1.2 que AWS fournit déjà par défaut.) Pour chiffrer les données de Session Manager session, créez une clé KMS symétrique à l'aide de. AWS KMS

AWS KMSle chiffrement est disponible pour `Standard_StreamInteractiveCommands`, et les types de `NonInteractiveCommands` session. Pour utiliser l'option permettant de chiffrer les données de session à l'aide d'une clé créée dans AWS KMS, la version 2.3.539.0 ou une version ultérieure de AWS Systems Manager SSM Agent doit être installée sur le nœud géré.

> **Note**
>
> Vous devez activer le chiffrement AWS KMS pour réinitialiser les mots de passe sur vos nœuds gérés à partir de la console AWS Systems Manager. Pour plus d'informations, consultez Réinitialisation d'un mot de passe sur un nœud géré.

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 Compte AWS doit vous fournir les autorisations nécessaires pour utiliser la clé.

Une fois que vous avez activé le chiffrement de clé KMS pour vos données de session, les utilisateurs qui démarrent les sessions et les nœuds gérés auxquels ils se connectent doivent avoir l'autorisation d'utiliser la clé. Vous fournissez l'autorisation d'utiliser la clé KMS avec Session Manager via des politiques AWS Identity and Access Management (IAM). Pour plus d'informations, consultez les rubriques suivantes :

- Ajoutez des autorisations AWS KMS pour les utilisateurs de votre compte : Exemple de stratégies IAM pour Session Manager.
- Ajoutez des autorisations AWS KMS pour les nœuds gérés de votre compte : Étape 2 : vérifier ou ajouter des autorisations d'instance pour Session Manager.

Pour plus d'informations sur la création et la gestion de clés KMS, veuillez consulter le Guide du développeur AWS Key Management Service.

Pour de plus amples informations sur l'utilisation de AWS CLI pour activer le chiffrement de clé KMS des données de session dans votre compte, veuillez consulter Création d'un document de préférences Session Manager (ligne de commande) ou Mettre à jour les préférences Session Manager (ligne de commande).

> **Note**
>
> L'utilisation de clés KMS entraîne des frais. Pour obtenir des informations, veuillez consulter Tarification AWS Key Management Service.

Pour activer le chiffrement des données de session par clé KMS (console)

1. Ouvrez la console AWS Systems Manager à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Session Manager.
3. Sélectionnez l'onglet Préférences, puis Modifier.
4. Activez la case à cocher en regard de Enable KMS encryption (Activer le chiffrement KMS).
5. Effectuez l'une des actions suivantes :

- Cliquez sur le bouton en regard de Select a KMS key in my current account (Sélectionner une clé dans mon compte actuel), puis sélectionnez une clé dans la liste.

  -ou-

  Sélectionnez le bouton en regard de Entrer un alias de clé KMS ou un ARN de clé KMS. Saisissez manuellement un alias de clé KMS 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-

  Sélectionnez Create new key (Créer une clé) pour créer une clé KMS 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 dans le Guide de l'utilisateur AWS Key Management Service.

6. Sélectionnez Enregistrer.

Création d'un document de préférences Session Manager (ligne de commande)

Utilisez la procédure suivante pour créer des documents SSM qui définissent vos préférences pour les AWS Systems Manager Session Manager sessions. Vous pouvez utiliser le document pour configurer les options de session, notamment le chiffrement des données, la durée de session et la journalisation. Par exemple, vous pouvez spécifier si vous souhaitez stocker les données du journal de session dans un bucket Amazon Simple Storage Service (Amazon S3) ou dans un groupe de journaux CloudWatch Amazon Logs. Vous pouvez créer des documents qui définissent des préférences générales pour toutes les sessions pour un Compte AWS et Région AWS, ou qui définissent des préférences pour des sessions individuelles.

> **Note**
>
> Vous pouvez également configurer les préférences générales de session à l'aide de la console Session Manager.

Les documents utilisés pour définir les préférences de Session Manager doivent comporter un `sessionType` `Standard_Stream`. Pour plus d'informations sur les documents de session, veuillez consulter la rubrique the section called “Schéma de document de session”.

Pour plus d'informations sur l'utilisation de la ligne de commande afin de mettre à jour les préférences Session Manager existantes, veuillez consulter la rubrique Mettre à jour les préférences Session Manager (ligne de commande).

Pour un exemple de création de préférences de session en utilisant AWS CloudFormation, consultez le document Create a Systems Manager pour les Session Manager préférences dans le Guide de AWS CloudFormation l'utilisateur.

> **Note**
>
> Cette procédure décrit comment créer des documents pour définir Session Manager les préférences au Compte AWS niveau. Pour créer des documents qui seront utilisés pour définir les préférences au niveau de la session, spécifiez une valeur autre que `SSM-SessionManagerRunShell` pour les entrées de commande liées au nom de fichier. Pour utiliser votre document afin de définir les préférences des sessions démarrées à partir de l' AWS Command Line Interface (AWS CLI), indiquez le nom du document comme valeur du paramètre `--document-name`. Pour définir les préférences des sessions démarrées depuis la console Session Manager, vous pouvez saisir ou sélectionner le nom de votre document dans une liste.

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 le fichier.

```
{
    "schemaVersion": "1.0",
    "description": "Document to hold regional settings for Session Manager",
```

```
    "sessionType": "Standard_Stream",
    "inputs": {
        "s3BucketName": "",
        "s3KeyPrefix": "",
        "s3EncryptionEnabled": true,
        "cloudWatchLogGroupName": "",
        "cloudWatchEncryptionEnabled": true,
        "cloudWatchStreamingEnabled": false,
        "kmsKeyId": "",
        "runAsEnabled": false,
        "runAsDefaultUser": "",
        "idleSessionTimeout": "",
        "maxSessionDuration": "",
        "shellProfile": {
            "windows": "date",
            "linux": "pwd;ls"
        }
    }
}
```

Vous pouvez aussi transmettre des valeurs à vos préférences de session en utilisant des paramètres plutôt qu'en codant les valeurs en dur, comme l'illustre l'exemple suivant.

```
{
   "schemaVersion":"1.0",
   "description":"Session Document Parameter Example JSON Template",
   "sessionType":"Standard_Stream",
   "parameters":{
      "s3BucketName":{
         "type":"String",
         "default":""
      },
      "s3KeyPrefix":{
         "type":"String",
         "default":""
      },
      "s3EncryptionEnabled":{
         "type":"Boolean",
         "default":"false"
      },
      "cloudWatchLogGroupName":{
         "type":"String",
         "default":""
```

```
      },
      "cloudWatchEncryptionEnabled":{
         "type":"Boolean",
         "default":"false"
      }
   },
   "inputs":{
      "s3BucketName":"{{s3BucketName}}",
      "s3KeyPrefix":"{{s3KeyPrefix}}",
      "s3EncryptionEnabled":"{{s3EncryptionEnabled}}",
      "cloudWatchLogGroupName":"{{cloudWatchLogGroupName}}",
      "cloudWatchEncryptionEnabled":"{{cloudWatchEncryptionEnabled}}",
      "kmsKeyId":""
   }
}
```

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 CloudWatch journaux de journaux. Si vous souhaitez continuer à chiffrer des données entre le client local et les nœuds gérés, fournissez la clé KMS à utiliser pour le chiffrement. Voici un exemple.

```
{
  "schemaVersion": "1.0",
  "description": "Document to hold regional settings for Session Manager",
  "sessionType": "Standard_Stream",
  "inputs": {
    "s3BucketName": "DOC-EXAMPLE-BUCKET",
    "s3KeyPrefix": "MyS3Prefix",
    "s3EncryptionEnabled": true,
    "cloudWatchLogGroupName": "MyLogGroupName",
    "cloudWatchEncryptionEnabled": true,
    "cloudWatchStreamingEnabled": false,
    "kmsKeyId": "MyKMSKeyID",
    "runAsEnabled": true,
    "runAsDefaultUser": "MyDefaultRunAsUser",
    "idleSessionTimeout": "20",
    "maxSessionDuration": "60",
    "shellProfile": {
        "windows": "MyCommands",
        "linux": "MyCommands"
    }
  }
}
```

> **Note**
>
> Si vous ne souhaitez pas chiffrer les données de journaux, remplacez `true` par `false` pour `s3EncryptionEnabled`.
> Si vous n'envoyez pas de journaux à un compartiment Amazon S3 ou à un groupe de CloudWatch journaux de journaux, si vous ne souhaitez pas chiffrer les données de session actives ou si vous ne souhaitez pas activer le support Run As pour les sessions de votre compte, vous pouvez supprimer les lignes correspondant à ces options. Assurez-vous que la dernière ligne de la section `inputs` ne se termine pas par une virgule.
> Si vous ajoutez un ID de clé KMS pour chiffrer vos données de session, les utilisateurs qui démarrent les sessions et les nœuds gérés auxquels ils se connectent doivent avoir l'autorisation d'utiliser la clé. Vous fournissez l'autorisation d'utiliser la clé KMS avec Session Manager via des politiques IAM. Pour plus d'informations, consultez les rubriques suivantes :
>
> - Ajoutez AWS KMS des autorisations pour les utilisateurs de votre compte : Exemple de stratégies IAM pour Session Manager
> - Ajoutez AWS KMS des autorisations pour les nœuds gérés dans votre compte : Étape 2 : vérifier ou ajouter des autorisations d'instance pour Session Manager

3. Enregistrez le fichier.
4. Dans le répertoire où vous avez créé le fichier JSON, exécutez la commande suivante.

   Linux & macOS

   ```
   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 votre outil de ligne de commande préféré pour modifier les AWS Systems Manager Session Manager préférences de Compte AWS la zone sélectionnée Région AWS. Utilisez Session Manager les préférences pour spécifier les options de journalisation des données de session dans un bucket Amazon Simple Storage Service (Amazon S3) ou un groupe

de journaux CloudWatch Amazon Logs. 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 le fichier.

```
{
    "schemaVersion": "1.0",
    "description": "Document to hold regional settings for Session Manager",
    "sessionType": "Standard_Stream",
    "inputs": {
        "s3BucketName": "",
        "s3KeyPrefix": "",
        "s3EncryptionEnabled": true,
        "cloudWatchLogGroupName": "",
        "cloudWatchEncryptionEnabled": true,
        "cloudWatchStreamingEnabled": false,
        "kmsKeyId": "",
        "runAsEnabled": true,
        "runAsDefaultUser": "",
        "idleSessionTimeout": "",
        "maxSessionDuration": "",
        "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 CloudWatch journaux de journaux. Si vous souhaitez chiffrer davantage les données entre le client local et les nœuds gérés, fournissez le code AWS KMS key à utiliser pour le chiffrement. Voici un exemple.

```
{
  "schemaVersion": "1.0",
  "description": "Document to hold regional settings for Session Manager",
  "sessionType": "Standard_Stream",
  "inputs": {
    "s3BucketName": "DOC-EXAMPLE-BUCKET",
```

```
    "s3KeyPrefix": "MyS3Prefix",
    "s3EncryptionEnabled": true,
    "cloudWatchLogGroupName": "MyLogGroupName",
    "cloudWatchEncryptionEnabled": true,
    "cloudWatchStreamingEnabled": false,
    "kmsKeyId": "MyKMSKeyID",
    "runAsEnabled": true,
    "runAsDefaultUser": "MyDefaultRunAsUser",
    "idleSessionTimeout": "20",
    "maxSessionDuration": "60",
    "shellProfile": {
        "windows": "MyCommands",
        "linux": "MyCommands"
    }
  }
}
```

> **Note**
>
> Si vous ne souhaitez pas chiffrer les données de journaux, remplacez `true` par `false` pour `s3EncryptionEnabled`.
> Si vous n'envoyez pas de journaux à un compartiment Amazon S3 ou à un groupe de CloudWatch journaux de journaux, si vous ne souhaitez pas chiffrer les données de session actives ou si vous ne souhaitez pas activer le support Run As pour les sessions de votre compte, vous pouvez supprimer les lignes correspondant à ces options. Assurez-vous que la dernière ligne de la section `inputs` ne se termine pas par une virgule.
> Si vous ajoutez un ID de clé KMS pour chiffrer vos données de session, les utilisateurs qui démarrent les sessions et les nœuds gérés auxquels ils se connectent doivent avoir l'autorisation d'utiliser la clé. Vous autorisez l'utilisation de la clé KMS Session Manager via des politiques AWS Identity and Access Management (IAM). Pour plus d'informations, consultez les rubriques suivantes :
>
> - Ajoutez AWS KMS des autorisations pour les utilisateurs de votre compte :Exemple de stratégies IAM pour Session Manager.
> - Ajoutez AWS KMS des autorisations pour les nœuds gérés dans votre compte :Étape 2 : vérifier ou ajouter des autorisations d'instance pour Session Manager.

3. Enregistrez le fichier.

4. Dans le répertoire où vous avez créé le fichier JSON, exécutez la commande suivante.

   Linux & macOS

   ```
   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 (facultative) : Restriction de l'accès aux commandes dans une session

Vous pouvez limiter les commandes qu'un utilisateur peut exécuter dans une AWS Systems Manager Session Manager session en utilisant un document de `Session` type personnalisé AWS Systems Manager (SSM). Dans ce document, vous définissez la commande qui est exécutée lorsque l'utilisateur démarre une session et les paramètres qu'il peut fournir à la commande. Le paramètre `Session` du document `schemaVersion` doit être défini sur 1.0 et le paramètre `sessionType` du document doit être défini sur `InteractiveCommands`. Vous pouvez ensuite créer des politiques AWS Identity and Access Management (IAM) permettant aux utilisateurs d'accéder uniquement aux documents `Session` que vous définissez. Pour de plus amples informations sur l'utilisation des politiques IAM pour restreindre l'accès aux commandes dans une session, consultez Exemples de politique IAM pour les commandes interactives.

Les documents marqués du signe `sessionType` de ne `InteractiveCommands` sont pris en charge que pour les sessions démarrées à partir du AWS Command Line Interface (AWS CLI). L'utilisateur fournit le nom du document personnalisé en tant que valeur du paramètre `--document-name` et fournit toutes les valeurs des paramètres de commande à l'aide de l'option `--parameters`. Pour de plus amples informations sur l'exécution des commandes interactives, consultez Démarrage d'une session (commandes interactives et non interactives).

Mettez en œuvre la procédure suivante pour 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 AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.
3. Sélectionnez Create command or session (Créer une commande ou une session).

4. Pour Name (Nom), saisissez un nom évocateur pour le document.
5. Pour Document type (Type de document), sélectionnez 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. Sélectionnez Créer un document.

Restreindre l'accès aux commandes dans une session (ligne de commande)

Avant de commencer

Si ce n'est pas déjà fait, installez et configurez le AWS Command Line Interface (AWS CLI) ou le AWS Tools for PowerShell. Pour plus d'informations, consultez la section Installation ou mise à jour de la version la plus récente de l' AWS CLI et Installation d' AWS Tools for PowerShell.

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 & macOS

   ```
   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"
```

Les paramètres de commande interactifs et le AWS CLI

Plusieurs façons permettent de définir des paramètres de commande interactifs lors de l'utilisation de la AWS CLI. En fonction du système d'exploitation (OS) de votre machine cliente que vous utilisez pour vous connecter aux nœuds gérés AWS CLI, la syntaxe que vous fournissez pour les commandes contenant des caractères spéciaux ou d'échappement peut être différente. Les exemples suivants montrent les différentes manières dont vous pouvez fournir des paramètres de commande lorsque vous utilisez le AWS CLI et comment gérer les caractères spéciaux ou d'échappement.

Les paramètres stockés dans Parameter Store peuvent être référencés dans AWS CLI les paramètres de votre commande, comme indiqué dans l'exemple suivant.

Linux & macOS

```
aws ssm start-session \
    --target instance-id \
    --document-name MyInteractiveCommandDocument \
    --parameters '{"command":["{{ssm:mycommand}}"]}'
```

Windows

```
aws ssm start-session ^
    --target instance-id ^
    --document-name MyInteractiveCommandDocument ^
    --parameters '{"command":["{{ssm:mycommand}}"]}'
```

L'exemple suivant montre l'utilisation d'une syntaxe abrégée avec la AWS CLI pour transmettre des paramètres.

Linux & macOS

```
aws ssm start-session \
    --target instance-id \
    --document-name MyInteractiveCommandDocument \
    --parameters command="ifconfig"
```

Windows

```
aws ssm start-session ^
    --target instance-id ^
    --document-name MyInteractiveCommandDocument ^
    --parameters command="ipconfig"
```

Vous pouvez également fournir des paramètres en JSON, comme illustré dans l'exemple suivant.

Linux & macOS

```
aws ssm start-session \
    --target instance-id \
    --document-name MyInteractiveCommandDocument \
    --parameters '{"command":["ifconfig"]}'
```

Windows

```
aws ssm start-session ^
    --target instance-id ^
    --document-name MyInteractiveCommandDocument ^
    --parameters '{"command":["ipconfig"]}'
```

Les paramètres peuvent également être stockés dans un fichier JSON et fournis au AWS CLI , comme indiqué dans l'exemple suivant. Pour de plus amples informations sur l'utilisation de paramètres de la AWS CLI à partir d'un fichier, consultez Chargement de paramètres AWS CLI à partir d'un fichier dans le Guide de l'utilisateur AWS Command Line Interface .

```
{
    "command": [
        "my command"
    ]
}
```

Linux & macOS

```
aws ssm start-session \
    --target instance-id \
    --document-name MyInteractiveCommandDocument \
```

```
    --parameters file://complete/path/to/file/parameters.json
```

Windows

```
aws ssm start-session ^
    --target instance-id ^
    --document-name MyInteractiveCommandDocument ^
    --parameters file://complete/path/to/file/parameters.json
```

Vous pouvez également générer un AWS CLI squelette à partir d'un fichier d'entrée JSON, comme illustré dans l'exemple suivant. Pour plus d'informations sur la génération de AWS CLI squelettes à partir de fichiers d'entrée JSON, consultez la section Génération de AWS CLI squelettes et de paramètres d'entrée à partir d'un fichier d'entrée JSON ou YAML dans le Guide de l'AWS Command Line Interface utilisateur.

```
{
    "Target": "instance-id",
    "DocumentName": "MyInteractiveCommandDocument",
    "Parameters": {
        "command": [
            "my command"
        ]
    }
}
```

Linux & macOS

```
aws ssm start-session \
    --cli-input-json file://complete/path/to/file/parameters.json
```

Windows

```
aws ssm start-session ^
    --cli-input-json file://complete/path/to/file/parameters.json
```

Lorsque des caractères d'échappement sont placés entre guillemets, vous devez ajouter des barres obliques inverses supplémentaires aux caractères d'échappement, comme l'illustre l'exemple suivant.

Linux & macOS

```
aws ssm start-session \
    --target instance-id \
    --document-name MyInteractiveCommandDocument \
    --parameters '{"command":["printf \"abc\\\\tdef\""]}'
```

Windows

```
aws ssm start-session ^
    --target instance-id ^
    --document-name MyInteractiveCommandDocument ^
    --parameters '{"command":["printf \"abc\\\\tdef\""]}'
```

Pour plus d'informations sur l'utilisation de guillemets avec les paramètres de commande dans la AWS CLI, consultez Utilisation de guillemets avec des chaînes dans la AWS CLI dans le Guide de l'utilisateur AWS Command Line Interface .

Exemples de politique IAM pour les commandes interactives

Vous pouvez créer des politiques 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 un seul nœud géré

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":"ssm:StartSession",
         "Resource":[
            "arn:aws:ec2:region:987654321098:instance/i-02573cafcfEXAMPLE",
            "arn:aws:ssm:region:987654321098:document/exampleAllowedSessionDocument"
         ],
         "Condition":{
            "BoolIfExists":{
               "ssm:SessionDocumentAccessCheck":"true"
            }
```

```
         }
      }
   ]
}
```

Autoriser un utilisateur à exécuter une commande interactive sur tous les nœuds gérés

```
{
   "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 tous les nœuds gérés

```
{
   "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 (facultative) : Utiliser AWS PrivateLink pour configurer un point de terminaison de VPC pour Session Manager

Vous pouvez renforcer la sécurité de vos nœuds gérés en configurant AWS Systems Manager pour qu'il utilise un point de terminaison Virtual Private Cloud (VPC) d'interface. Les points de terminaison de l'interface sont alimentés par AWS PrivateLink une technologie qui vous permet d'accéder en privé aux API Amazon Elastic Compute Cloud (Amazon EC2) et Systems Manager à l'aide d'adresses IP privées.

AWS PrivateLink restreint tout le trafic réseau entre vos nœuds gérés, Systems Manager et Amazon EC2 vers le réseau Amazon. (Les nœuds gérés n'ont pas accès à Internet). De même, vous n'avez pas besoin d'une passerelle Internet, d'un périphérique NAT ni d'une passerelle réseau privé virtuel.

Pour plus d'informations sur la création d'un point de terminaison VPC, consultez Améliorer la sécurité des instances EC2 en utilisant des points de terminaison VPC pour Systems Manager.

L'alternative à l'utilisation d'un point de terminaison de VPC est l'activation de l'accès Internet sortant sur vos nœuds gérés. Dans ce cas, les nœuds gérés 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 Manager utilise le dernier de ces points de terminaison, `ssmmessages.region.amazonaws.com`, pour appeler le service Session Manager dans le cloud à partir de SSM Agent.

Pour utiliser des fonctionnalités facultatives telles que le chiffrement AWS Key Management Service (AWS KMS), le streaming des CloudWatch journaux vers Amazon CloudWatch Logs (Logs) et l'envoi

de journaux vers Amazon Simple Storage Service (Amazon S3), vous devez autoriser le trafic sortant HTTPS (port 443) vers les points de terminaison suivants :

- `kms.region.amazonaws.com`
- `logs.region.amazonaws.com`
- `s3.region.amazonaws.com`

Pour de plus amples informations sur les points de terminaison requis pour Systems Manager, veuillez consulter Référence : ec2messages, ssmmessages et autres opérations d'API.

Étape 7 : (Facultatif) activez ou désactivez les autorisations administratives du compte ssm-user.

À partir de la version 2.3.50.0 de l'AWS Systems Manager SSM Agent, l'agent crée un compte utilisateur local appelé `ssm-user` et l'ajoute à `/etc/sudoers` (Linux et macOS) ou au groupe Administrateurs (Windows). Sur les versions de l'agent antérieures à 2.3.612.0, le compte est créé la première fois que l'SSM Agent 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 un nœud. `ssm-user` correspond à l'utilisateur du système d'exploitation par défaut lorsqu'une session AWS Systems Manager Session Manager est démarrée. La version SSM Agent 2.3.612.0 est sortie le 8 mai 2019.

Si vous souhaitez interdire aux utilisateurs Session Manager d'exécuter des commandes administratives sur un nœud, vous pouvez mettre à jour les autorisations du compte `ssm-user`. Vous pouvez restaurer ces autorisations après qu'elles ont été supprimées.

Rubriques

- Gestion des autorisations de compte sudo ssm-user sous Linux et macOS
- Gérer les autorisations administrateur de compte ssm-user sur Windows Server

Gestion des autorisations de compte sudo ssm-user sous Linux et macOS

Utilisez l'une des procédures suivantes pour activer ou désactiver les autorisations de compte sudo ssm-user sur les nœuds gérés Linux et macOS.

Utiliser Run Command pour modifier les autorisations sudo ssm-user (console)

- Exécutez la procédure décrite sur la page Exécution des commande à partir de la console en appliquant les valeurs suivantes :
  - Pour Command document (Document de commande), sélectionnez `AWS-RunShellScript`.
  - Pour supprimer un accès sudo, dans la zone Command parameters (Paramètres de la commande), collez la commande suivante dans la zone Commands (Commandes) :

    ```
    cd /etc/sudoers.d
    echo "#User rules for ssm-user" > ssm-agent-users
    ```

    -ou-

    Pour restaurer un accès sudo, dans la zone Command parameters (Paramètres de la commande), collez la commande suivante dans 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 au nœud géré 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. Sauvegardez le fichier.

Gérer les autorisations administrateur de compte ssm-user sur Windows Server

Utilisez l'une des procédures suivantes pour activer ou désactiver les autorisations administrateur de compte ssm-user sur les nœuds gérés Windows Server.

Utiliser Run Command pour modifier les autorisations administrateur (console)

- Exécutez la procédure décrite sur la page Exécution des commande à partir de la console en appliquant les valeurs suivantes :

  Pour Command document (Document de commande), sélectionnez `AWS-RunPowerShellScript`.

  Pour supprimer un accès administrateur, dans la zone Command parameters (Paramètres de la commande), collez la commande suivante dans la zone Commands (Commandes).

  ```
  net localgroup "Administrators" "ssm-user" /delete
  ```

  -ou-

  Pour restaurer un accès administrateur, dans la zone Command parameters (Paramètres de la commande), collez la commande suivante dans la zone Commands (Commandes).

  ```
  net localgroup "Administrators" "ssm-user" /add
  ```

Utilisation de la fenêtre d'invite de commande ou PowerShell pour modifier les autorisations administrateur

1. Connectez-vous au nœud géré et ouvrez PowerShell ou la fenêtre d'invite de commande.
2. Pour supprimer un accès administrateur, exécutez la commande suivante.

   ```
   net localgroup "Administrators" "ssm-user" /delete
   ```

-ou-

Pour restaurer un accès administrateur, exécutez la commande suivante.

```
net localgroup "Administrators" "ssm-user" /add
```

Utilisation de la console Windows pour modifier les autorisations administrateur

1. Connectez-vous au nœud géré et ouvrez PowerShell ou la fenêtre d'invite de commande.
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 administrateur, saisissez **`Administrators`** dans la zone de texte, puis sélectionnez Add (Ajouter).
5. Sélectionnez OK.

Étape 8 : (Facultatif) Autoriser et contrôler les autorisations pour les connexions SSH via Session Manager

Vous pouvez autoriser les utilisateurs de votre compte Compte AWS à utiliser le AWS Command Line Interface (AWS CLI) pour établir des connexions Secure Shell (SSH) aux nœuds gérés à l'aide AWS Systems Manager Session Manager de. Les utilisateurs qui se connectent à l'aide de SSH peuvent également copier des fichiers entre leurs ordinateurs locaux et les nœuds gérés à l'aide de SCP (Secure Copy Protocol). Vous pouvez utiliser cette fonctionnalité pour vous connecter aux nœuds gérés sans avoir à ouvrir de ports entrants ou à maintenir des hôtes bastions.

Après avoir autorisé les connexions SSH, vous pouvez utiliser des politiques AWS Identity and Access Management (IAM) pour autoriser ou refuser explicitement aux utilisateurs, aux groupes ou aux rôles d'établir des connexions SSH à l'aide de ceux-ci. Session Manager

> **Note**
>
> La journalisation n'est pas disponible pour les sessions Session Manager qui se connectent via le réacheminement de port ou SSH. Cela est dû au fait que SSH chiffre toutes les données de session et que Session Manager sert uniquement de tunnel pour les connexions SSH.

Rubriques

Autorisation des connexions SSH pour Session Manager

Suivez les étapes suivantes pour autoriser les connexions SSH via Session Manager sur un nœud géré.

Pour autoriser des connexions SSH pour Session Manager

1. Sur le nœud géré vers lequel vous souhaitez activer les connexions SSH, procédez comme suit :

   - Assurez-vous que SSH s'exécute sur le nœud géré. (Vous pouvez fermer les ports entrants sur le nœud.)
   - Assurez-vous que SSM Agent 2.3.672.0 ou une version ultérieure est installé sur le nœud géré.

     Pour plus d'informations sur l'installation ou la mise à jour de SSM Agent sur un nœud géré, consultez les rubriques suivantes :

     - Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Windows Server.
     - Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Linux
     - Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour macOS
     - Comment installer le SSM Agent sur des nœuds Windows hybrides
     - Comment installer le SSM Agent sur des nœuds Linux hybrides

> **Note**
>
> Pour utiliser Session Manager avec des serveurs et des machines virtuelles sur site activés en tant que nœuds gérés, utilisez le niveau d'instances avancées. Pour de plus amples informations, sur les instances avancées, veuillez consulter Configuration des niveaux d'instance.

2. Sur l'ordinateur local à partir duquel vous souhaitez vous connecter à un nœud géré 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 Installez le Session Manager plugin pour AWS CLI.

   - 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 et macOS

     > **Tip**
     >
     > Le fichier de configuration SSH est généralement situé dans `~/.ssh/config`.

     Ajoutez le fichier de configuration suivant 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 le fichier de configuration suivant 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"
```

- Vérifiez que vous disposez d'un certificat au format PEM (Privacy Enhanced Mail) ou au minimum d'une clé publique, à utiliser lors de l'établissement de connexions avec les nœuds gérés. Il doit s'agir d'une clé déjà associée au nœud géré. Les autorisations de votre fichier de clé privée doivent être configurés afin que vous soyez le ou la seul(e) à pouvoir le lire. Vous pouvez utiliser la commande suivante pour définir les autorisations de votre fichier de clé privée afin que vous soyez le ou la seul(e) à pouvoir le lire.

  ```
  chmod 400 <my-key-pair>.pem
  ```

  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 de session. Pour plus d'informations sur le démarrage d'une session à l'aide de SSH, consultez Démarrage d'une session (SSH).)

Contrôle des autorisations utilisateur pour des connexions SSH en utilisant Session Manager

Après avoir activé des connexions SSH sur un nœud géré en utilisant Session Manager, vous pouvez utiliser des politiques IAM pour autoriser ou empêcher des utilisateurs, des groupes ou des rôles d'établir des connexions SSH en utilisant Session Manager.

Pour utiliser une politique IAM afin d'autoriser des connexions SSH en utilisant Session Manager

- Utilisez l'une des options suivantes :

- Option 1 : ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.

  Dans le panneau de navigation, sélectionnez Policies (Politiques), puis mettez à jour la politique d'autorisations permettant à l'utilisateur ou au rôle de votre choix de démarrer les connexions SSH via Session Manager.

  Par exemple, ajoutez l'élément suivant à la politique Quickstart que vous avez créée dans Démarrage rapide - Politiques d'utilisateur final pour Session Manager. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ssm:StartSession",
            "Resource": [
                "arn:aws:ec2:region:account-id:instance/instance-id",
                "arn:aws:ssm:*:*:document/AWS-StartSSHSession"
            ],
            "Condition": {
                "BoolIfExists": {
                    "ssm:SessionDocumentAccessCheck": "true"
                }
             }
        }
    ]
}
```

- Option 2 : associez une politique en ligne à une politique utilisateur à l'aide de l'API AWS Management Console AWS CLI, de ou de l' AWS API.

  À l'aide de la méthode de votre choix, associez la déclaration de politique de l'option 1 à la politique d'un AWS utilisateur, d'un groupe ou d'un rôle.

  Pour de plus amples informations, veuillez consulter Ajout et suppression d'autorisations basées sur l'identité IAM dans le Guide de l'utilisateur IAM.

Pour utiliser une politique IAM afin de refuser des connexions SSH en utilisant Session Manager

- Utilisez l'une des options suivantes :

  - Option 1 : ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/. Dans le panneau de navigation, sélectionnez Policies (Politiques), puis mettez à jour la politique d'autorisations pour l'utilisateur ou le rôle afin de bloquer le démarrage des sessions Session Manager.

    Par exemple, ajoutez l'élément suivant à la politique Quickstart que vous avez créée dans Démarrage rapide - Politiques d'utilisateur final pour Session Manager.

    ```
    {
        "Version": "2012-10-17",
        "Statement": [
            {
                "Sid": "VisualEditor1",
                "Effect": "Deny",
                "Action": "ssm:StartSession",
                "Resource": "arn:aws:ssm:*:*:document/AWS-StartSSHSession"
            }
        ],
                "Condition": {
                    "BoolIfExists": {
                        "ssm:SessionDocumentAccessCheck": "true"
                    }
                 }
    }
    ```

  - Option 2 : associez une politique en ligne à une politique utilisateur à l'aide de l'API AWS Management Console AWS CLI, de ou de l' AWS API.

    À l'aide de la méthode de votre choix, associez la déclaration de politique de l'option 1 à la politique d'un AWS utilisateur, d'un groupe ou d'un rôle.

    Pour de plus amples informations, veuillez consulter Ajout et suppression d'autorisations basées sur l'identité IAM dans le Guide de l'utilisateur IAM.

Utilisation des Session Manager

Vous pouvez utiliser la console AWS Systems Manager, la console Amazon Elastic Compute Cloud (Amazon EC2) ou la AWS Command Line Interface (AWS CLI) pour démarrer des sessions qui vous connectent aux nœuds gérés auxquels votre administrateur système vous a octroyé l'accès à l'aide de politiques 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. Une fois qu'une session est établie, elle n'est pas affectée par la durée de la session de rôle IAM. Pour plus d'informations sur la limitation de la durée de la session avec Session Manager, consultez Spécifier une valeur de délai d'expiration d'une session inactive et Spécification de la durée de session maximale.

Pour de plus amples informations sur les sessions, veuillez consulter la page Qu'est-ce qu'une session ?.

Rubriques

Installez le Session Manager plugin pour AWS CLI

Pour lancer des sessions Session Manager avec vos nœuds gérés en utilisant la AWS Command Line Interface (AWS CLI), vous devez d'abord installer le plug-in Session Manager sur votre machine locale. Vous pouvez installer le plug-in sur les versions prises en charge de Microsoft Windows ServermacOS, Linux etUbuntu Server.

> **Note**
>
> Pour utiliser le Session Manager plugin, la AWS CLI version 1.16.12 ou ultérieure doit être installée sur votre machine locale. Pour plus d'informations, consultez Installation ou mise à jour de la version la plus récente de l' AWS Command Line Interface.

Rubriques

- Installer le plugin Session Manager sur Windows
- Installer le plugin Session Manager sur macOS
- Installez le Session Manager plugin sur Amazon Linux 2 et ses Red Hat Enterprise Linux distributions
- Installer le plugin Session Manager sur Debian Server et Ubuntu Server
- Vérifier l'Installation du plugin Session Manager
- Session Managerplugin activé GitHub
- (Facultatif) Activer la journalisation du plugin Session Manager

Dernière version et historique des versions du plugin Session Manager

Votre ordinateur local doit exécuter une version prise en charge du plugin Session Manager. 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 vérifier que vous disposez bien 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 plugin Session Manager, ainsi que les fonctionnalités et les améliorations incluses dans chaque version.

Version	Date de publication	Détails
1,2.633.0	30 mai 2024	Amélioration : mise à jour du Dockerfile pour utiliser une image Amazon Elastic Container Registry (Amazon ECR).

Version	Date de publication	Détails
1,2,553.0	10 janvier 2024	Amélioration : packages Golang améliorés aws-sdk-go et dépendants.
1,2.536,0	4 décembre 2023	Amélioration : Ajout de la prise en charge de la transmiss ion d'une réponse d'StartSessionAPI en tant que variable d'environnement à session-manager-plugin.
1,2,497,0	1er août 2023	Amélioration : mise à niveau du kit SDK Go vers la version 1.44.302.
1,2,463,0	15 mars 2023	Amélioration : Ajout de la Mac with Apple silicon prise en charge d'Apple Mac (M1) dans le programme d'installation du bundle macOS et dans le programme d'installation signé.
1,2,398,0	14 octobre 2022	Amélioration : prise en charge de la version 1.17 de golang. Mettez à jour le session-manager-plugin lanceur par défaut pour macOS afin d'utiliser python3. Mettez à jour le chemin d'importation de SSMCLI vers. session-manager-plugin
1,2,339,0	16 juin 2022	Correctif de bogue : correction du délai d'expiration de la session inactive pour les sessions de port.
1,2.331.0	27 mai 2022	Correctif de bogue : correction des sessions de port se fermant prématurément lorsque le serveur local ne se connecte pas avant le délai d'attente.
1,2.323.0	19 mai 2022	Correctif de bogue: Désactivation de la fonctionnalité smux keep alive pour utiliser la fonctionnalité idle session timeout.
1,2,312,0	31 mars 2022	Amélioration : prend en charge plus de types de charges utiles de messages de sortie.
1,2.295,0	12 janvier 2022	Bug fix (Correction de bogue) : sessions bloquées en raison du renvoi des données de flux par le client lorsque l'agent devient inactif, et des journaux incorrects pour les messages `start_publication` et `pause_publication` .

Version	Date de publication	Détails
1,2,279,0	27 octobre 2021	Amélioration : emballage zip pour la plateforme Windows.
1.2.245.0	19 août 2021	Amélioration : mettez `aws-sdk-go` à la dernière version (v1.40.17) pour prendre en charge AWS IAM Identity Center.
1.2.234.0	26 juillet 2021	Correction de bogue : une session Handle a brusquement interrompu le scénario dans un type de session interactive.
1.2.205,0	10 juin 2021	Amélioration : ajout de la prise en charge du programme d'installation macOS signé.
1.2.54.0	29 janvier 2021	Amélioration : Ajout de la prise en charge de l'exécution de sessions en mode NonInteractiveCommands exécution.
1,2,30,0	24 novembre 2020	Amélioration : (sessions de réacheminement de port uniquement) Amélioration de la performance globale.
1.2.7.0	15 octobre 2020	Amélioration : (sessions de réacheminement de port uniquement) réduction de la latence et amélioration de la performance globale.
1.1.61.0	17 avril 2020	Amélioration : ajout de la prise en charge ARM pour Linux et Ubuntu.
1.1.54.0	6 janvier 2020	Correction de bogue : gérer le scénario de condition de course des paquets abandonnés lorsque le plugin Session Manager n'est pas prêt.
1.1.50.0	19 novembre 2019	Amélioration: Ajout du support pour le transfert d'un port vers un socket unix local.
1.1.35.0	7 novembre 2019	Amélioration : (sessions de transfert de port uniquement) Envoyez une TerminateSession commande SSM Agent lorsque l'utilisateur local appuie sur`Ctrl+C`.

Version	Date de publication	Détails
1.1.33.0	26 septembre 2019	Amélioration : (sessions de réacheminement de port uniquement) Envoie d'un signal de déconnexion au serveur lorsque le client abandonne la connexion TCP.
1.1.31.0	6 septembre 2019	Amélioration : mise à jour pour maintenir 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 de Session Manager.
1.1.17.0	4 avril 2019	Amélioration : ajout de la prise en charge d'autres données de chiffrement de session avec AWS Key Management Service (AWS KMS).
1.0.37.0	20 septembre 2018	Amélioration : correction de bogue pour la version Windows.
1.0.0.0	11 septembre 2018	Version initiale du plugin Session Manager.

Installer le plugin Session Manager sur Windows

Vous pouvez installer le plug-in Session Manager sur 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 du plugin Session Manager.

> **Note**
>
> Pour obtenir de meilleurs résultats, nous vous recommandons de démarrer des sessions sur les clients Windows à l'aide de Windows PowerShell version 5 ou ultérieure. Vous pouvez également utiliser le shell Command sur Windows 10. Le plug-in Session Manager prend

uniquement en charge PowerShell et le shell Command. Les outils de ligne de commande tiers peuvent ne pas être compatibles avec le plugin.

Pour installer le plugin Session Manager 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
   ```

   Vous pouvez également télécharger une version zippée du programme d'installation à l'aide de l'URL suivante.

   ```
   https://s3.amazonaws.com/session-manager-downloads/plugin/latest/windows/SessionManagerPlugin.zip
   ```

2. Exécutez le programme d'installation que vous avez téléchargé et suivez les instructions à l'écran. Si vous avez téléchargé la version zippée du programme d'installation, vous devez d'abord décompresser le programme d'installation.

   Laissez vide la zone Emplacement d'installation pour installer le plugin dans le répertoire par défaut.

   - `%PROGRAMFILES%\Amazon\SessionManagerPlugin\bin\`

3. Vérifiez que l'installation a réussi. Pour plus d'informations, consultez Vérifier l'Installation du plugin Session Manager.

   > **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 Plug-in Session Manager pas ajouté automatiquement au chemin de la ligne de commande (Windows).

Installer le plugin Session Manager sur macOS

Sélectionnez l'une des rubriques suivantes pour installer le plug-in Session Manager sur macOS. Le programme d'installation fourni utilise un fichier ZIP. Une fois décompressé, vous pouvez installer le plugin à l'aide du binaire. Le programme d'installation signé est un fichier .pkg signé.

Rubriques

Installer le plugin Session Manager sur macOS

Cette section explique comment installer le plug-in Session Manager sur macOS en utilisant le 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 plugin Session Manager à l'aide du programme d'installation fourni (macOS)

1. Téléchargez le programme d'installation.

   x86_64

   ```
   curl "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/mac/
   sessionmanager-bundle.zip" -o "sessionmanager-bundle.zip"
   ```

   Mac avec silicone Apple

   ```
   curl "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/
   mac_arm64/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 plugin nécessite la version 2.6.5 ou ultérieure de Python, ou la version 3.3 ou ultérieure de Python. 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 plugin Session Manager, exécutez le script d'installation avec cette version dans le chemin d'accès absolu au fichier exécutable Python. Voici un exemple.
>
> ```
> sudo /usr/local/bin/python3.8 sessionmanager-bundle/install -i /usr/local/
> sessionmanagerplugin -b /usr/local/bin/session-manager-plugin
> ```

Le programme d'installation installe le plugin Session Manager 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 afficher une description des options `-i` et `-b`, utilisez l'option `-h`.

```
./sessionmanager-bundle/install -h
```

4. Vérifiez que l'installation a réussi. Pour plus d'informations, consultez Vérifier l'Installation du plugin Session Manager.

> **Note**
>
> Pour 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 plugin Session Manager sur macOS avec le programme d'installation signé

Cette section explique comment installer le plug-in Session Manager sur macOS en utilisant le programme d'installation signé.

Pour installer le plugin Session Manager en utilisant le programme d'installation signé (macOS)

1. Téléchargez le programme d'installation signé.

   x86_64

   ```
   curl "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/mac/
   session-manager-plugin.pkg" -o "session-manager-plugin.pkg"
   ```

   Mac avec silicone Apple

   ```
   curl "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/
   mac_arm64/session-manager-plugin.pkg" -o "session-manager-plugin.pkg"
   ```

2. Exécutez les commandes d'installation.

   ```
   sudo installer -pkg session-manager-plugin.pkg -target /
   sudo ln -s /usr/local/sessionmanagerplugin/bin/session-manager-plugin /usr/local/
   bin/session-manager-plugin
   ```

3. Vérifiez que l'installation a réussi. Pour plus d'informations, consultez Vérifier l'Installation du plugin Session Manager.

Installez le Session Manager plugin sur Amazon Linux 2 et ses Red Hat Enterprise Linux distributions

Utilisez la procédure suivante pour installer le plugin Session Manager sur les distributions RHEL.

> **Note**
>
> Le Session Manager plugin n'est pas pris en charge sur Amazon Linux 1. Il est pris en charge sur Amazon Linux 2 et versions ultérieures.

1. Téléchargez et installez le package RPM du plug-in Session Manager.

x86_64

Sur RHEL 7, exécutez la commande suivante :

```
sudo yum install -y https://s3.amazonaws.com/session-manager-downloads/plugin/
latest/linux_64bit/session-manager-plugin.rpm
```

Sur les RHEL versions 8 et 9, exécutez la commande suivante :

```
sudo dnf install -y https://s3.amazonaws.com/session-manager-downloads/plugin/
latest/linux_64bit/session-manager-plugin.rpm
```

x86

Sur RHEL 7, exécutez la commande suivante :

```
sudo yum install -y https://s3.amazonaws.com/session-manager-downloads/plugin/
latest/linux_32bit/session-manager-plugin.rpm
```

Sur les RHEL versions 8 et 9, exécutez la commande suivante :

```
sudo dnf install -y https://s3.amazonaws.com/session-manager-downloads/plugin/
latest/linux_32bit/session-manager-plugin.rpm
```

ARM64

Sur RHEL 7, exécutez la commande suivante :

```
sudo yum install -y https://s3.amazonaws.com/session-manager-downloads/plugin/
latest/linux_arm64/session-manager-plugin.rpm
```

Sur les RHEL versions 8 et 9, exécutez la commande suivante :

```
sudo dnf install -y https://s3.amazonaws.com/session-manager-downloads/plugin/
latest/linux_arm64/session-manager-plugin.rpm
```

2. Vérifiez que l'installation a réussi. Pour plus d'informations, veuillez consulter Vérifier l'Installation du plugin Session Manager.

> **Note**
>
> Si vous souhaitez désinstaller le plug-in, exécutez `sudo yum erase session-manager-plugin -y`

Installer le plugin Session Manager sur Debian Server et Ubuntu Server

1. Téléchargez le package deb du plugin Session Manager.

   x86_64

   ```
   curl "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/
   ubuntu_64bit/session-manager-plugin.deb" -o "session-manager-plugin.deb"
   ```

   x86

   ```
   curl "https://s3.amazonaws.com/session-manager-downloads/plugin/latest/
   ubuntu_32bit/session-manager-plugin.deb" -o "session-manager-plugin.deb"
   ```

   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, veuillez consulter Vérifier l'Installation du plugin Session Manager.

> **Note**
>
> Si vous souhaitez désinstaller le plug-in, exécutez `sudo dpkg -r session-manager-plugin`

Vérifier l'Installation du plugin Session Manager

Exécutez les commandes suivantes pour vérifier que le plugin Session Manager a bien été installé.

```
session-manager-plugin
```

Si l'installation a réussi, vous obtenez le message suivant.

```
The Session Manager plugin is installed successfully. Use the AWS CLI to start a
 session.
```

Vous pouvez également tester l'installation en exécutant la start-sessioncommande dans le AWS Command Line Interface(AWS CLI). Dans la commande ci-après, remplacez *instance-id* avec vos propres informations.

```
aws ssm start-session --target instance-id
```

Cette commande ne fonctionnera que si vous avez installé et configuré le AWS CLI, et si votre Session Manager administrateur vous a accordé les autorisations IAM nécessaires pour accéder au nœud géré cible à l'aide Session Manager de.

Session Managerplugin activé GitHub

Le code source du Session Manager plugin est disponible sur GitHubafin que vous puissiez adapter le plugin à vos besoins. Nous vous conseillons d'envoyer des requêtes d'extraction pour les modifications que vous souhaitez inclure. Toutefois, Amazon Web Services ne fournit pas de support pour l'exécution de copies modifiées de ce logiciel.

(Facultatif) Activer la journalisation du plugin Session Manager

Le plugin Session Manager 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 plugin Session Manager 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

- Activation de la journalisation pour le plug-in Session Manager (Windows)
- Activation de la journalisation pour le plug-in Session Manager (Linux et macOS)

Activation de la journalisation pour le plug-in Session Manager (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 journaux sur l'ouverture d'un canal de données et les reconnexions aux sessions 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 de `formatid="fmtinfo"` par `formatid="fmtdebug"`.
   - Options des fichiers journaux : vous pouvez modifier les options des fichiers journaux, y compris l'emplacement de stockage des journaux, 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:\Program Files\Amazon\SessionManagerPlugin
     \Logs\session-manager-plugin.log" maxsize="30000000" maxrolls="5"/>
     <filter levels="error,critical" formatid="fmterror">
     <rollingfile type="size" filename="C:\Program Files\Amazon\SessionManagerPlugin
     \Logs\errors.log" maxsize="10000000" maxrolls="5"/>
     ```

5. Sauvegardez le fichier.

Activation de la journalisation pour le plug-in Session Manager (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 journaux sur l'ouverture de canaux de données et les reconnexions aux sessions 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 de `formatid="fmtinfo"` par `outputs formatid="fmtdebug"`.
   - Options des fichiers journaux : vous pouvez modifier les options des fichiers journaux, y compris l'emplacement de stockage des journaux, 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. Sauvegardez le fichier.

Démarrer une session

Vous pouvez utiliser la AWS Systems Manager console, la console Amazon Elastic Compute Cloud (Amazon EC2), AWS Command Line Interface le AWS CLI() ou SSH pour démarrer une session.

Rubriques

- Démarrage d'une session (console Systems Manager)
- Démarrage d'une session (console Amazon EC2)
- Démarrage d'une session (AWS CLI)
- Démarrage d'une session (SSH)
- Démarrage d'une session (réacheminement de port)
- Démarrage d'une session (réacheminement de port vers un hôte distant)
- Démarrage d'une session (commandes interactives et non interactives)

Démarrage d'une session (console Systems Manager)

Vous pouvez utiliser la AWS Systems Manager console pour démarrer une session avec un nœud géré dans votre compte.

> **Note**
>
> Avant de démarrer une session, vérifiez que vous bien effectué les étapes de configuration pour Session Manager. Pour plus d'informations, consultez Configuration de Session Manager.

Pour démarrer une session (console Systems Manager)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Session Manager.
3. Sélectionnez Start session (Démarrer une session).
4. (Facultatif) Saisissez une description de session dans le champ Raison de la session.
5. Dans la liste Instances cibles, sélectionnez la case d'option située à gauche du nœud géré auquel vous souhaitez vous connecter.

   Si le nœud que vous souhaitez ne figure pas dans la liste, ou si vous sélectionnez un nœud et recevez une erreur de configuration, veuillez consulter Nœud géré non disponible ou non configuré pour Session Manager pour connaître les étapes de résolution du problème.
6. Choisissez Démarrer la session pour lancer la session immédiatement.

   -ou-

   Choisissez Suivant pour connaître les options de session.
7. (Facultatif) Dans Document de session, sélectionnez le document que vous souhaitez exécuter au démarrage de la session. Si votre document prend en charge les paramètres d'exécution, vous pouvez saisir une ou plusieurs valeurs séparées par des virgules dans chaque champ de paramètre.
8. Choisissez Suivant.
9. Sélectionnez Start session (Démarrer une session).

Une fois la connexion effectuée, vous pouvez exécuter des commandes bash (Linux et macOS) ou des commandes PowerShell (Windows) comme vous le feriez via n'importe quel autre type de connexion.

> **Important**
>
> Si vous souhaitez autoriser les utilisateurs à spécifier un document lorsqu'ils démarrent des sessions dans la console Session Manager, tenez compte des points suivants :
>
> - Vous devez accorder aux utilisateurs les autorisations `ssm:GetDocument` et `ssm:ListDocuments` dans leur politique IAM. Pour plus d'informations, consultez Octroi de l'accès aux documents de session personnalisés dans la console.

- La console prend uniquement en charge les documents de session dont le `sessionType` est défini sur `Standard_Stream`. Pour plus d'informations, consultez Schéma de document de session.

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 vos informations de connexion. Demandez à cette personne de mettre à jour vos politiques afin de vous autoriser à démarrer des sessions à partir de la console Amazon EC2. Si vous êtes administrateur, consultez Exemple de stratégies IAM pour Session Manager 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, sélectionnez Instances.
3. Sélectionnez l'instance, puis sélectionnez Connect (Connexion).
4. Pour Méthode de connexion, sélectionnez Session Manager.
5. Sélectionnez Connexion.

Une fois la connexion effectuée, vous pouvez exécuter des commandes bash (Linux et macOS) ou des commandes PowerShell (Windows) comme vous le feriez via n'importe quel autre type de connexion.

Démarrage d'une session (AWS CLI)

Installez et configurez le AWS Command Line Interface (AWS CLI), si ce n'est pas déjà fait.

Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l' AWS CLI.

Avant de démarrer une session, vérifiez que vous bien effectué les étapes de configuration pour Session Manager. Pour plus d'informations, consultez Configuration de Session Manager.

Pour utiliser les commandes AWS CLI d'exécution de session, le Session Manager plugin doit également être installé sur votre machine locale. Pour plus d'informations, veuillez consulter Installez le Session Manager plugin pour AWS CLI.

Pour démarrer une session à l'aide de AWS CLI, exécutez la commande suivante en remplaçant *instance-id* par vos propres informations.

```
aws ssm start-session \
    --target instance-id
```

Pour plus d'informations sur les autres options que vous pouvez utiliser avec la start-session commande, reportez-vous start-sessionà la AWS Systems Manager section de la référence des AWS CLI commandes.

Démarrage d'une session (SSH)

Pour démarrer une session SSH Session Manager, SSM Agent version 2.3.672.0 ou version ultérieure doit être installé sur le nœud géré.

Conditions préalables pour une connexion SSH

Prenez note des exigences et limitations suivantes pour les connexions de session à l'aide de SSH :

- Votre nœud géré cible doit être configurée pour prendre en charge les connexions SSH. Pour plus d'informations, voir (facultatif) Autoriser et contrôler les autorisations pour les connexions SSH via Session Manager.
- Vous devez vous connecter en utilisant le compte du nœud géré associé au certificat PEM (Privacy Enhanced Mail), et non le compte `ssm-user` utilisé pour d'autres types de connexions de session. Par exemple, sur les instances EC2 pour Linux et macOS, 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 guide de l'utilisateur Amazon EC2.
- La journalisation n'est pas disponible pour les sessions Session Manager qui se connectent via le réacheminement de port ou SSH. Cela est dû au fait que SSH chiffre toutes les données de session et que Session Manager sert uniquement de tunnel pour les connexions SSH.

> **Note**
>
> Avant de démarrer une session, vérifiez que vous bien effectué les étapes de configuration pour Session Manager. Pour plus d'informations, consultez Configuration de Session Manager.

Pour démarrer une session à l'aide de SSH, exécutez la commande suivante. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

```
ssh -i /path/my-key-pair.pem username@instance-id
```

> **Tip**
>
> Lorsque vous démarrez une session à l'aide de SSH, vous pouvez copier des fichiers locaux sur le nœud géré cible en utilisant le format de commande suivant.
>
> ```
> scp -i /path/my-key-pair.pem /path/ExampleFile.txt username@instance-id:~
> ```

Pour plus d'informations sur les autres options que vous pouvez utiliser avec la start-session commande, reportez-vous start-sessionà la AWS Systems Manager section de la référence des AWS CLI commandes.

Démarrage d'une session (réacheminement de port)

Pour démarrer une session de réacheminement de port Session Manager, SSM Agent version 2.3.672.0 ou ultérieure doit être installé sur le nœud géré.

> **Note**
>
> Avant de démarrer une session, vérifiez que vous bien effectué les étapes de configuration pour Session Manager. Pour plus d'informations, consultez Configuration de Session Manager.
> Pour utiliser les commandes AWS CLI d'exécution de session, vous devez installer le Session Manager plugin sur votre machine locale. Pour plus d'informations, veuillez consulter Installez le Session Manager plugin pour AWS CLI.

Selon votre système d'exploitation et votre outil de ligne de commande, le placement des guillemets peut différer et des caractères d'échappement peuvent être nécessaires.

Pour démarrer une session de réacheminement de port, exécutez la commande suivante à partir de la CLI. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

Linux & macOS

```
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"
```

`portNumber`est le port distant du nœud géré où vous souhaitez que le trafic de session soit redirigé. Vous pouvez par exemple spécifier le port `3389` pour la connexion à un nœud Windows via le protocole RDP (Remote Desktop Protocol). Si vous ne spécifiez pas le paramètre `portNumber`, Session Manager utilise `80` comme valeur par défaut.

`localPortNumber`est le port de votre ordinateur local où le trafic commence, par exemple`56789`. Cette valeur est celle que vous saisissez lors de la connexion à un nœud géré en utilisant un client. Par exemple, **`localhost:56789`**.

Pour plus d'informations sur les autres options que vous pouvez utiliser avec la start-session commande, reportez-vous start-sessionà la AWS Systems Manager section de la référence des AWS CLI commandes.

Pour de plus amples informations sur les sessions de réacheminement de port, veuillez consulter réacheminement de port avec AWS Systems ManagerSession Manager dans le Blog d'actualitéAWS .

Démarrage d'une session (réacheminement de port vers un hôte distant)

Pour démarrer une session de transfert de port Session Manager vers un hôte distant, la version 3.1.1374.0 ou ultérieure de SSM Agent doit être installée sur le nœud géré. L'hôte distant ne doit pas nécessairement être géré par Systems Manager.

> **Note**
>
> Avant de démarrer une session, vérifiez que vous bien effectué les étapes de configuration pour Session Manager. Pour plus d'informations, consultez Configuration de Session Manager.
> Pour utiliser les commandes AWS CLI d'exécution de session, vous devez installer le Session Manager plugin sur votre machine locale. Pour plus d'informations, veuillez consulter Installez le Session Manager plugin pour AWS CLI.
> Selon votre système d'exploitation et votre outil de ligne de commande, le placement des guillemets peut différer et des caractères d'échappement peuvent être nécessaires.

Pour démarrer une session de redirection de port, exécutez la commande suivante à partir du AWS CLI. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

Linux & macOS

```
aws ssm start-session \
    --target instance-id \
    --document-name AWS-StartPortForwardingSessionToRemoteHost \
    --parameters '{"host":["mydb.example.us-east-2.rds.amazonaws.com"],"portNumber":
["3306"], "localPortNumber":["3306"]}'
```

Windows

```
aws ssm start-session ^
    --target instance-id ^
    --document-name AWS-StartPortForwardingSessionToRemoteHost ^
    --parameters host="mydb.example.us-
east-2.rds.amazonaws.com",portNumber="3306",localPortNumber="3306"
```

La valeur `host` représente le nom d'hôte ou l'adresse IP de l'hôte distant auquel vous souhaitez vous connecter. Les exigences générales de connectivité et de résolution de noms entre le nœud géré et l'hôte distant s'appliquent toujours.

`portNumber`est le port distant du nœud géré où vous souhaitez que le trafic de session soit redirigé. Vous pouvez par exemple spécifier le port `3389` pour la connexion à un nœud Windows via le protocole RDP (Remote Desktop Protocol). Si vous ne spécifiez pas le paramètre `portNumber`, Session Manager utilise `80` comme valeur par défaut.

`localPortNumber`est le port de votre ordinateur local où le trafic commence, par exemple`56789`. Cette valeur est celle que vous saisissez lors de la connexion à un nœud géré en utilisant un client. Par exemple, **`localhost:56789`**.

Pour plus d'informations sur les autres options que vous pouvez utiliser avec la start-session commande, reportez-vous start-sessionà la AWS Systems Manager section de la référence des AWS CLI commandes.

Démarrage d'une session avec une tâche Amazon ECS

Session Managerpermet de démarrer une session de redirection de port avec une tâche au sein d'un cluster Amazon Elastic Container Service (Amazon ECS). Pour ce faire, vous devez mettre à jour le rôle de tâche dans IAM afin d'inclure les autorisations suivantes :

```
{
   "Version": "2012-10-17",
   "Statement": [
       {
       "Effect": "Allow",
       "Action": [
            "ssmmessages:CreateControlChannel",
            "ssmmessages:CreateDataChannel",
            "ssmmessages:OpenControlChannel",
            "ssmmessages:OpenDataChannel"
       ],
      "Resource": "*"
      }
   ]
}
```

Pour démarrer une session de transfert de port avec une tâche Amazon ECS, exécutez la commande suivante à partir du AWS CLI. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

> **Note**
>
> Supprimez les **<** and **>** symboles du `target` paramètre. Ces symboles ne sont fournis qu'à titre de clarification pour le lecteur.

Linux & macOS

```
aws ssm start-session \
    --target ecs:<ECS_cluster_name>_<ECS_container_ID>_<container_runtime_ID> \
    --document-name AWS-StartPortForwardingSessionToRemoteHost \
    --parameters '{"host":["URL"],"portNumber":["port_number"], "localPortNumber":
["port_number"]}'
```

Windows

```
aws ssm start-session ^
    --target ecs:<ECS_cluster_name>_<ECS_container_ID>_<container_runtime_ID> ^
    --document-name AWS-StartPortForwardingSessionToRemoteHost ^
    --parameters host="URL",portNumber="port_number",localPortNumber="port_number"
```

Démarrage d'une session (commandes interactives et non interactives)

Avant de démarrer une session, vérifiez que vous bien effectué les étapes de configuration pour Session Manager. Pour plus d'informations, consultez Configuration de Session Manager.

Pour utiliser les commandes AWS CLI d'exécution de session, le Session Manager plugin doit également être installé sur votre machine locale. Pour plus d'informations, veuillez consulter Installez le Session Manager plugin pour AWS CLI.

Pour démarrer une session de commande interactive, exécutez la commande suivante. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

Linux & macOS

```
aws ssm start-session \
    --target instance-id \
    --document-name CustomCommandSessionDocument \
    --parameters '{"logpath":["/var/log/amazon/ssm/amazon-ssm-agent.log"]}'
```

Windows

```
aws ssm start-session ^
    --target instance-id ^
    --document-name CustomCommandSessionDocument ^
    --parameters logpath="/var/log/amazon/ssm/amazon-ssm-agent.log"
```

Pour plus d'informations sur les autres options que vous pouvez utiliser avec la start-session commande, reportez-vous start-sessionà la AWS Systems Manager section de la référence des AWS CLI commandes.

Plus d'informations

- Utiliser la redirection de port AWS Systems ManagerSession Manager pour se connecter à des hôtes distants
- Redirection de port d'instance Amazon EC2 avec AWS Systems Manager
- Gérez les ressources Microsoft AD AWS gérées avec la redirection de Session Manager port
- Port Forwarding Using AWS Systems ManagerSession Manager sur AWS News Blog.

Résilier une session

Vous pouvez utiliser la console AWS Systems Manager ou l'AWS Command Line Interface (AWS CLI) pour mettre fin à une session que vous avez démarrée dans 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

- Terminer une session (console)
- Résiliation d'une session (AWS CLI)

Terminer une session (console)

Vous pouvez utiliser la console AWS Systems Manager pour démarrer une session 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, sélectionnez Session Manager.
3. Pour Sessions, sélectionnez le bouton d'option situé à gauche de la session que vous souhaitez résilier.
4. Sélectionnez Terminer.

Résiliation d'une session (AWS CLI)

Pour terminer une session à l'aide de l'AWS CLI, exécutez la commande suivante. Remplacez *session-id* avec vos propres informations.

```
aws ssm terminate-session \
    --session-id session-id
```

Pour de plus amples informations sur la commande terminate-session, veuillez consulter terminate-session dans la section AWS Systems Manager de la Référence des commandes de la AWS CLI.

Afficher l'historique de session

Vous pouvez utiliser la console AWS Systems Manager ou l'AWS Command Line Interface (AWS CLI) pour afficher des informations sur les sessions de votre compte. Dans la console, vous pouvez par exemple afficher les informations suivantes :

- L'ID de la session
- Quel utilisateur s'est connecté à un nœud géré au cours de la session
- ID du nœud géré
- 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 de journalisation, consultez la page Activation et désactivation de la journalisation des activités de session.

Rubriques

- Afficher l'historique de session (console)
- Affichage de l'historique de session (AWS CLI)

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, sélectionnez Session Manager.
3. Sélectionnez l'onglet Session history (Historique de session).

   -ou-

   Si la page d'accueil de Session Manager s'ouvre en premier, choisissez Configurer les préférences, puis l'onglet Historique des sessions.

Affichage de l'historique de session (AWS CLI)

Pour afficher la liste des sessions de votre compte à l'aide de l'AWS CLI, 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 que vous pouvez utiliser avec la commande describe-sessions, consultez describe-sessions dans la section AWS Systems Manager de la Référence de Command AWS CLI.

Auditer 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 la possibilité d'auditer l'activité de votre compte Compte AWS à l'aide de AWS CloudTrail.

CloudTrail capture les appels d'API de session via la console Systems Manager, le AWS Command Line Interface (AWS CLI) et le SDK Systems Manager. Vous pouvez consulter les informations sur la CloudTrail console ou les stocker dans un compartiment Amazon Simple Storage Service (Amazon S3) spécifique. Un compartiment Amazon S3 est utilisé pour tous les CloudTrail journaux de votre compte. Pour plus d'informations, consultez Journalisation des appels d' AWS Systems Manager API avec AWS CloudTrail.

> **Note**
>
> Pour une analyse analytique récurrente et historique de vos fichiers journaux, pensez à interroger les CloudTrail journaux à l'aide de CloudTrail Lake ou d'une table que vous gérez. Pour plus d'informations, consultez la section Interrogation des AWS CloudTrail journaux dans le guide de AWS CloudTrail l'utilisateur.

Surveillance de l'activité des sessions à l'aide d'Amazon EventBridge (console)

Avec EventBridge, vous pouvez définir des règles pour détecter les modifications apportées aux 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.

EventBridge le support de Session Manager repose sur les enregistrements des opérations d'API enregistrés par CloudTrail. (Vous pouvez utiliser CloudTrail l'intégration avec EventBridge pour

répondre à la plupart des AWS Systems Manager événements.) Les actions effectuées au cours d'une session, telles qu'une `exit` commande, qui n'effectuent pas d'appel d'API ne sont pas détectées par EventBridge.

Les étapes suivantes expliquent comment lancer des notifications via Amazon Simple Notification Service (Amazon SNS) lorsqu'un événement d'API Session Manager se produit, tel que StartSession.

Pour surveiller l'activité des sessions à l'aide d'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 en savoir plus, consultez Création d'une rubrique dans le Manuel du développeur d'Amazon Simple Notification Service.

2. Créez une EventBridge règle pour appeler la cible Amazon SNS pour le type d'Session Managerévénement que vous souhaitez suivre.

   Pour plus d'informations sur la création de la règle, consultez la section Création de EventBridge règles Amazon qui réagissent aux événements dans le guide de EventBridge l'utilisateur Amazon.

   Tout au long de la création de votre règle, sélectionnez les éléments suivants :

   - Pour le AWS service choisissez Systems Manager.
   - Pour Type d'événement, choisissez AWS API Call through 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, ResumeSession et TerminateSession. (EventBridge ne prend pas en charge `Describe*` les commandes `Get* List*`, et.)
   - Pour Sélectionner une cible, choisissez Rubrique SNS. Pour Topic (Rubrique), sélectionnez le nom de la rubrique Amazon SNS que vous avez créée à l'étape 1.

Pour plus d'informations, consultez le guide de EventBridge l'utilisateur Amazon et le guide de démarrage d'Amazon Simple Notification Service.

Activation et désactivation de la journalisation des activités 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 dans votre compte 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 nœuds gérés via Session Manager au cours des 30 derniers jours.
- Générez des notifications relatives à l'activité de votre session Compte AWS, telles que les notifications Amazon Simple Notification Service (Amazon SNS).
- Lancez automatiquement une autre action sur une AWS ressource à la suite d'une activité de session, telle que l'exécution d'une AWS Lambda fonction, le démarrage d'un AWS CodePipeline pipeline ou l'exécution d'un AWS Systems Manager Run Command document.

Important

Prenez note des exigences et limitations suivantes pour Session Manager :

- Session Manager journalise les commandes que vous saisissez, et leur sortie, durant une session, en fonction de vos préférences de session. Pour empêcher l'affichage de données sensibles, telles que les mots de passe, dans vos journaux de session, nous vous recommandons d'utiliser les commandes suivantes lors de la saisie de données sensibles durant une session.

  Linux & macOS

  ```
  stty -echo; read passwd; stty echo;
  ```

  Windows

  ```
  $Passwd = Read-Host -AsSecureString
  ```

- 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 nœuds gérés Linux ou macOS, assurez-vous que l'utilitaire d'écran est installé. Si ce n'est pas le cas, il se peut que vos données de journaux soient tronquées. Sur Amazon Linux 1, Amazon Linux 2, AL2023 etUbuntu Server, l'utilitaire d'écran est installé par défaut. Pour installer l'écran manuellement, selon votre version de Linux, exécutez `sudo yum install screen` ou `sudo apt-get install screen`.
- La journalisation n'est pas disponible pour les sessions Session Manager qui se connectent via le réacheminement de port ou SSH. Cela est dû au fait que SSH chiffre 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 la journalisation des données de session, consultezCréation d'un rôle IAM avec des autorisations pour Session Manager Amazon S3 et CloudWatch Logs (console).

Pour de plus amples informations sur les options de journalisation pour Session Manager, veuillez consulter les rubriques suivantes.

Rubriques

- Données de session de streaming à l'aide d'Amazon CloudWatch Logs (console)
- Journalisation des données de session avec Amazon S3 (console)
- Enregistrement des données de session à l'aide d'Amazon CloudWatch Logs (console)
- Désactivation de la journalisation des Session Manager activités dans CloudWatch Logs et Amazon S3

Données de session de streaming à l'aide d'Amazon CloudWatch Logs (console)

Vous pouvez envoyer un flux continu de journaux de données de session à Amazon CloudWatch Logs. Les détails essentiels, tels que les commandes qu'un utilisateur a exécutées dans une session, l'ID de l'utilisateur qui a exécuté les commandes et les horodatages indiquant le moment où les données de session sont diffusées vers CloudWatch Logs, sont inclus lors de la diffusion en continu des données de session. Lors du streaming de données de session, les journaux sont au format JSON afin de faciliter l'intégration à vos solutions de journalisation existantes. Le streaming de données de session n'est pas pris en charge pour les commandes interactives

> **Note**
>
> Le streaming de données de session à partir de nœuds gérés Windows Server implique que PowerShell 5.1 ou version ultérieure soit installé. Par défaut, la version PowerShell requise est installée sur Windows Server 2016 et versions ultérieures. Cependant, Windows Server 2012 et 2012 R2 n'ont pas la version PowerShell requise installée par défaut. Si vous n'avez pas encore mis à jour PowerShell sur vos nœuds gérés Windows Server 2012 ou 2012 R2, vous pouvez le faire en utilisant Run Command. Pour plus d'informations sur la mise à jour de PowerShell à l'aide de Run Command, veuillez consulter la rubrique Mise à jour PowerShell en utilisant Run Command.

> **Important**
>
> Si le paramètre de politique de PowerShell transcription est configuré sur vos nœuds Windows Server gérés, vous ne pourrez pas diffuser les données de session.

Pour diffuser des données de session à l'aide d'Amazon CloudWatch Logs (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Session Manager.
3. Sélectionnez l'onglet Préférences, puis Modifier.
4. Cochez la case à côté de Activer dans le cadre de la CloudWatch journalisation.
5. Sélectionnez l'option Streaming de journaux de session.
6. (Recommandé) Cochez la case à côté de Autoriser uniquement les groupes de CloudWatch journaux chiffrés. Une fois cette option activée, les données de journaux sont chiffrées à l'aide du chiffrement côté serveur des clés spécifiées pour le groupe de journaux. Si vous ne souhaitez pas chiffrer les données du journal envoyées à CloudWatch Logs, décochez la case. Vous devez également décocher la case si le chiffrement n'est pas autorisé sur le groupe de journaux.
7. Pour les CloudWatch journaux, pour spécifier le groupe de CloudWatch journaux Logs existant dans le répertoire Compte AWS auquel vous souhaitez télécharger les journaux de session, sélectionnez l'une des options suivantes :

- Saisissez le nom d'un groupe de journaux dans la zone de texte qui a déjà été créé dans votre compte pour stocker les données de journal de session.
- Browse log groups (Rechercher des groupes de journaux) : sélectionnez un groupe de journaux qui a déjà été créé dans votre compte pour stocker les données de journaux de session.

8. Sélectionnez Enregistrer.

Journalisation des données de session avec Amazon S3 (console)

Vous pouvez choisir de stocker les données des journaux de session dans un compartiment Amazon Simple Storage Service (Amazon S3) de votre choix à des fins de débogage et de résolution des problèmes. L'option par défaut est pour les journaux à envoyer à un compartiment Amazon S3 chiffré. Le chiffrement est effectué à l'aide de la clé spécifiée pour le compartiment, qu'il s' AWS KMS key agisse d'une clé de chiffrement côté serveur (SSE) Amazon S3 (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 Amazon S3

Pour envoyer les journaux vers votre compartiment Amazon S3 via le chiffrement, le chiffrement doit être autorisé sur le compartiment. Pour de plus amples informations sur le chiffrement des compartiments Amazon S3, consultez la section Chiffrement par défaut Amazon S3 pour les compartiments S3.

Clé gérée par le client

Si vous utilisez une clé KMS que vous gérez vous-même pour chiffrer votre compartiment, alors le profil d'instance IAM attaché à vos instances doit disposer des autorisations explicites pour lire la clé. Si vous utilisez un Clé gérée par AWS, l'instance n'a pas besoin de cette autorisation explicite. Pour de plus amples informations sur l'octroi d'une autorisation d'utilisation d'une clé à un profil d'instance,

veuillez consulter Autorise les utilisateurs de clé à utiliser la clé dans le Guide du développeur AWS Key Management Service .

Suivez ces étapes pour configurer Session Manager afin de stocker les journaux de session dans un compartiment Amazon S3.

> **Note**
>
> Vous pouvez également utiliser le AWS CLI pour spécifier ou modifier le compartiment Amazon S3 auquel les données de session sont envoyées. Pour plus d'informations, veuillez consulter Mettre à jour les préférences Session Manager (ligne de commande).

Pour journaliser les données de session avec Amazon S3 (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Session Manager.
3. Sélectionnez l'onglet Préférences, puis Modifier.
4. Cochez la case en regard de Activer, sous Journalisation S3.
5. (Recommandé) Cochez la case en regard de Allow only encrypted S3 buckets (Autoriser uniquement les compartiments S3 chiffrés). Lorsque cette option est activée, les données de journaux sont chiffrées à l'aide de la clé de chiffrement côté serveur spécifiée pour le compartiment. Si vous ne voulez pas chiffrer les données de journaux qui sont envoyées vers Amazon S3, décochez la case. Vous devez également décocher la case si le chiffrement n'est pas autorisé 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 de plus amples informations sur les conventions de dénomination des compartiments Amazon S3, veuillez consulter Limites et restrictions applicables aux compartiments dans le Guide de l'utilisateur Amazon Simple Storage Service.

- Choose a bucket name from the list (Choisir un nom de compartiment dans la liste) : sélectionnez un compartiment Amazon S3 qui a déjà été créé dans votre compte pour stocker les données de journaux de session.
- Saisissez le nom du compartiment dans la zone de texte : saisissez le nom d'un compartiment Amazon S3 qui a déjà été créé dans votre compte pour stocker les données de journal 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. Sélectionnez Enregistrer.

Pour plus d'informations sur l'utilisation d'Amazon S3 et des compartiments Amazon S3, veuillez consulter le Guide de l'utilisateur Amazon Simple Storage Service et le Guide de l'utilisateur Amazon Simple Storage Service.

Enregistrement des données de session à l'aide d'Amazon CloudWatch Logs (console)

Avec Amazon CloudWatch Logs, vous pouvez surveiller, stocker et accéder à des fichiers journaux provenant de différents sites Services AWS. Vous pouvez envoyer les données du journal de session à un groupe de CloudWatch journaux de journaux à des fins de débogage et de résolution des problèmes. L'option par défaut est que les données de journaux soient envoyées avec chiffrement à l'aide de votre clé KMS, mais vous pouvez envoyer les données à votre groupe de journaux avec ou sans chiffrement.

Suivez ces étapes AWS Systems Manager Session Manager pour configurer l'envoi des données du journal de session à un groupe de CloudWatch journaux de journaux à la fin de vos sessions.

> **Note**
>
> Vous pouvez également utiliser le AWS CLI pour spécifier ou modifier le groupe de CloudWatch journaux des journaux auquel les données de session sont envoyées. Pour plus d'informations, veuillez consulter Mettre à jour les préférences Session Manager (ligne de commande).

Pour enregistrer les données de session à l'aide d'Amazon CloudWatch Logs (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Session Manager.
3. Sélectionnez l'onglet Préférences, puis Modifier.
4. Cochez la case à côté de Activer dans le cadre de la CloudWatch journalisation.
5. Sélectionnez l'option Chargement de journaux de session.
6. (Recommandé) Cochez la case à côté de Autoriser uniquement les groupes de CloudWatch journaux chiffrés. Une fois cette option activée, les données de journaux sont chiffrées à l'aide du chiffrement côté serveur des clés spécifiées pour le groupe de journaux. Si vous ne souhaitez pas chiffrer les données du journal envoyées à CloudWatch Logs, décochez la case. Vous devez également décocher la case si le chiffrement n'est pas autorisé sur le groupe de journaux.
7. Pour les CloudWatch journaux, pour spécifier le groupe de CloudWatch journaux Logs existant dans le répertoire Compte AWS auquel vous souhaitez télécharger les journaux de session, sélectionnez l'une des options suivantes :
   - Choose a log group from the list (Choisir un groupe de journaux dans la liste) : utilisez 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 : entrez le nom d'un groupe de journaux qui a déjà été créé dans votre compte pour stocker les données de journal de session.
8. Sélectionnez Enregistrer.

Pour plus d'informations sur l'utilisation des CloudWatch journaux, consultez le guide de l'utilisateur Amazon CloudWatch Logs.

Désactivation de la journalisation des Session Manager activités dans CloudWatch Logs et Amazon S3

Vous pouvez utiliser la console Systems Manager ou AWS CLI désactiver la journalisation des activités de session dans votre compte.

Pour désactiver la journalisation des activités de session (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Session Manager.
3. Sélectionnez l'onglet Préférences, puis Modifier.
4. Pour désactiver la CloudWatch journalisation, dans la section de CloudWatch journalisation, décochez la case Activer.
5. Pour désactiver la journalisation S3, décochez la case Activer dans la section Journalisation S3.
6. Choisissez Enregistrer.

Pour désactiver la journalisation des activités de session (AWS CLI)

Pour désactiver la journalisation des activités de session à l'aide du AWS CLI, suivez les instructions figurant dansMettre à jour les préférences Session Manager (ligne de commande).

Dans votre fichier JSON, assurez-vous que les entrées `s3BucketName` et `cloudWatchLogGroupName` ne contiennent aucune valeur. Par exemple :

```
"inputs": {
        "s3BucketName": "",
        ...
        "cloudWatchLogGroupName": "",
        ...
    }
```

Vous pouvez également supprimer toutes les entrées `S3*` et `cloudWatch*` de votre fichier JSON pour désactiver le journal.

Schéma de document de session

Les informations suivantes décrivent les éléments du schéma d'un document de session. AWS Systems Manager Session Managerutilise les documents de session pour déterminer le type de session à démarrer, par exemple une session standard, une session de transfert de port ou une session pour exécuter une commande interactive.

schemaVersion

Version de schéma du document de session. Les documents de session prennent uniquement en charge la version 1.0.

Type : chaîne

Obligatoire : oui

description

Description que vous spécifiez pour le document de session. Par exemple, « Document pour démarrer la session de réacheminement de port avec Session Manager ».

Type : chaîne

Obligatoire : non

sessionType

Type de session établie en utilisant le document de session.

Type : chaîne

Obligatoire : oui

Valeurs valides : `InteractiveCommands` | `NonInteractiveCommands` | `Port` | `Standard_Stream`

inputs

Préférences de session à utiliser pour les sessions établies en utilisant ce document de session. Cet élément est nécessaire pour les documents de session utilisés pour créer des sessions `Standard_Stream`.

Type : StringMap

Obligatoire : non

s3BucketName

Le compartiment Amazon Simple Storage Service (Amazon S3) vers lequel vous voulez envoyer des journaux de session au terme de vos sessions.

Type : chaîne

Obligatoire : non

s3KeyPrefix

Préfixe à utiliser lors de l'envoi de journaux au compartiment Amazon S3 que vous avez spécifié dans l'entrée `s3BucketName`. Pour de plus amples informations sur l'utilisation d'un préfixe partagé avec des objets stockés dans Amazon S3, veuillez consulter Utilisation d'un compartiment S3 dans le Guide de l'utilisateur Amazon Simple Storage.

Type : chaîne

Obligatoire : non

s3EncryptionEnabled

Si l'entrée est définie sur `true`, le compartiment Amazon S3 que vous avez spécifié dans l'entrée `s3BucketName` doit être chiffré.

Type : booléen

Obligatoire : oui

cloudWatchLogGroupName

Le nom du groupe Amazon CloudWatch Logs (CloudWatch Logs) auquel vous souhaitez envoyer les journaux de session à la fin de vos sessions.

Type : chaîne

Obligatoire : non

cloudWatchEncryptionEnabled

Si l'entrée est définie sur `true`, le groupe de journaux que vous avez spécifié dans l'entrée `cloudWatchLogGroupName` doit être chiffré.

Type : booléen

Obligatoire : oui

cloudWatchStreamingEnabled

Si l'entrée est définie sur `true`, un flux continu de journaux de données de session est envoyé au groupe de journaux que vous avez spécifié dans l'entrée `cloudWatchLogGroupName`. Si l'entrée est définie sur `false`, les journaux de session sont envoyés au groupe de journaux que vous avez spécifié dans l'entrée `cloudWatchLogGroupName` à la fin de vos sessions.

Type : booléen

Obligatoire : oui

kmsKeyId

L'ID que AWS KMS key vous souhaitez utiliser pour chiffrer davantage les données entre vos machines clientes locales et les nœuds gérés par Amazon Elastic Compute Cloud (Amazon EC2) auxquels vous vous connectez.

Type : chaîne

Obligatoire : non

runAsEnabled

Si l'entrée est définie sur `true`, vous devez spécifier un compte utilisateur existant sur les nœuds gérés auxquels vous vous connecterez dans l'entrée `runAsDefaultUser`. Sinon, les sessions ne démarreront pas. Par défaut, les sessions sont démarrées en utilisant le compte `ssm-user` créé par l'SSM Agent AWS Systems Manager . La fonction Exécuter en tant que n'est prise en charge que pour la connexion aux nœuds gérés Linux.

Type : booléen

Obligatoire : oui

runAsDefaultUser

Nom du compte utilisateur avec lequel démarrer les sessions sur les nœuds gérés Linux lorsque l'entrée `runAsEnabled` est définie sur `true`. Le compte utilisateur que vous spécifiez pour cette entrée doit exister sur les nœuds gérés auxquels vous vous connecterez ; sinon, les sessions ne démarreront pas.

Type : chaîne

Obligatoire : non

idleSessionTimeout

Durée d'inactivité que vous voulez autoriser avant la fin d'une session. Cette entrée est mesurée en minutes.

Type : chaîne

Valeurs valides : 1-60

Obligatoire : non

maxSessionDuration

Durée maximale que vous voulez autoriser avant la fin d'une session. Cette entrée est mesurée en minutes.

Type : chaîne

Valeurs valides : 1 à 1 440

Obligatoire : non

shellProfile

Préférences que vous spécifiez par système d'exploitation, et qui sont à appliquer dans les sessions, telles que les préférences de shell, les variables d'environnement, les répertoires de travail et l'exécution de plusieurs commandes au démarrage d'une session.

Type : StringMap

Obligatoire : non

windows

Préférences de shell, variables d'environnement, répertoires de travail et commandes que vous spécifiez pour les sessions sur les nœuds gérés Windows.

Type : chaîne

Obligatoire : non

linux

Préférences de shell, variables d'environnement, répertoires de travail et commandes que vous spécifiez pour les sessions sur les nœuds gérés Linux.

Type : chaîne

Obligatoire : non

parameters

Objet qui définit les paramètres acceptés par le document. Pour plus d'informations sur la définition des paramètres de document, consultez Paramètres dans le Éléments de données

niveau supérieur. Pour les paramètres que vous référencez souvent, nous vous recommandons de les stocker dans Parameter Store de Systems Manager, 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 paramètre Parameter Store en utilisant le format suivant.

```
{{ssm:parameter-name}}
```

Pour plus d'informations sur Parameter Store, consultez AWS Systems Manager Parameter Store.

Type : StringMap

Obligatoire : non

properties

Objet dont les valeurs que vous spécifiez sont utilisées dans l'opération d'API `StartSession`.

Pour les documents de session utilisés pour des sessions `InteractiveCommands`, l'objet Propriétés inclut les commandes à exécuter sur les systèmes d'exploitation que vous spécifiez. Vous pouvez également déterminer si les commandes sont exécutées en tant que `root` à l'aide de la propriété booléenne `runAsElevated`. Pour de plus amples informations, consultez Restreindre l'accès aux commandes dans une session.

Pour les documents de session utilisés pour des sessions `Port`, l'objet Propriétés contient le numéro de port vers lequel le trafic doit être redirigé. Pour obtenir un exemple, veuillez consulter l'exemple de document de session de type `Port` plus loin dans cette rubrique.

Type : StringMap

Obligatoire : non

Exemple de document de session de type `Standard_Stream`

YAML

```
---
```

```
schemaVersion: '1.0'
description: Document to hold regional settings for Session Manager
sessionType: Standard_Stream
inputs:
  s3BucketName: ''
  s3KeyPrefix: ''
  s3EncryptionEnabled: true
  cloudWatchLogGroupName: ''
  cloudWatchEncryptionEnabled: true
  cloudWatchStreamingEnabled: true
  kmsKeyId: ''
  runAsEnabled: true
  runAsDefaultUser: ''
  idleSessionTimeout: '20'
  maxSessionDuration: '60'
  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,
        "cloudWatchStreamingEnabled": true,
        "kmsKeyId": "",
        "runAsEnabled": true,
        "runAsDefaultUser": "",
        "idleSessionTimeout": "20",
        "maxSessionDuration": "60",
        "shellProfile": {
            "windows": "date",
            "linux": "pwd;ls"
        }
    }
}
```

Exemple de document de session de type `InteractiveCommands`

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
        }
    }
}
```

Exemple de document de session de type `Port`

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"
    }
}
```

Exemple de document de session avec caractères spéciaux

YAML

```
---
schemaVersion: '1.0'
description: Example document with quotation marks
sessionType: InteractiveCommands
parameters:
  Test:
    type: String
    description: Test Input
```

```
    maxChars: 32
properties:
  windows:
    commands: |
        $Test = '{{ Test }}'
        $myVariable = \"Computer name is $env:COMPUTERNAME\"
        Write-Host "Test variable: $myVariable`.`nInput parameter: $Test"
    runAsElevated: false
```

JSON

```
{
   "schemaVersion":"1.0",
   "description":"Test document with quotation marks",
   "sessionType":"InteractiveCommands",
   "parameters":{
      "Test":{
         "type":"String",
         "description":"Test Input",
         "maxChars":32
      }
   },
   "properties":{
      "windows":{
         "commands":[
            "$Test = '{{ Test }}'",
            "$myVariable = \\\"Computer name is $env:COMPUTERNAME\\\"",
            "Write-Host \"Test variable: $myVariable`.`nInput parameter: $Test\""
         ],
         "runAsElevated":false
      }
   }
}
```

Résolution des problèmes de Session Manager

Consultez les informations suivantes pour tenter de résoudre les problèmes liés à AWS Systems Manager Session Manager.

Rubriques

- Session Manager ne peut pas se connecter à partir de la console Amazon EC2

- Je n'ai pas le droit de démarrer une session
- Je n'ai pas le droit de modifier les préférences de session
- Nœud géré non disponible ou non configuré pour Session Manager
- Plugin Session Manager introuvable
- Plug-in Session Manager pas ajouté automatiquement au chemin de la ligne de commande (Windows)
- Le plugin Session Manager ne répond pas
- TargetNotConnecté
- Affichage d'un écran vide après le démarrage d'une session
- Le nœud géré cesse de répondre lorsque les sessions durent longtemps
- Une erreur s'est produite (InvalidDocument) lors de l'appel de l' StartSession opération

Session Manager ne peut pas se connecter à partir de la console Amazon EC2

Problème : après avoir créé une nouvelle instance, l'onglet Session Manager de la console Amazon Elastic Compute Cloud (Amazon EC2) ne vous donne pas la possibilité de vous connecter.

Solution A : créer un profil d'instance : si vous ne l'avez pas déjà fait (comme indiqué dans les informations figurant dans l'onglet Gestionnaire de session de la console EC2), créez un profil d'instance AWS Identity and Access Management (IAM) en utilisant. Quick Setup Quick Setupest une capacité de AWS Systems Manager.

Session Manager nécessite un profil d'instance IAM pour se connecter à votre instance. Vous pouvez créer un profil d'instance et l'attribuer à votre instance en créant une configuration de gestion des hôtes avec Quick Setup. Une configuration de gestion des hôtes crée un profil d'instance avec les autorisations requises et l'attribue à votre instance. Une configuration de gestion des hôtes active également d'autres fonctionnalités de Systems Manager et crée des rôles IAM pour exécuter ces fonctionnalités. L'utilisation de Quick Setup ou des fonctionnalités activées par la configuration de gestion des hôtes est gratuite. Ouvrez Quick Setup et créez une configuration de gestion des hôtes.

> **Important**
>
> Une fois que vous avez créé la configuration de gestion des hôtes, Amazon EC2 peut prendre plusieurs minutes pour enregistrer la modification et actualiser l'onglet Gestionnaire de session. Si l'onglet n'affiche aucun bouton Connect au bout de deux minutes, redémarrez

votre instance. Après le redémarrage, si vous ne voyez toujours pas l'option de connexion, ouvrez la Configuration rapide et vérifiez que vous n'avez qu'une seule configuration de gestion d'hôte. S'il y en a deux, supprimez la configuration la plus ancienne et patientez quelques minutes.

Si vous ne parvenez toujours pas à vous connecter après avoir créé une configuration de gestion des hôtes, ou si vous recevez un message d'erreur, notamment un message d'erreur concernant SSM Agent, consultez l'une des solutions suivantes :

- Solution B : aucune erreur, mais impossible de se connecter
- Solution C : erreur concernant l'absence de l'SSM Agent

Solution B : aucune erreur, mais impossible de se connecter

Si vous avez créé la configuration de gestion des hôtes, que vous avez attendu plusieurs minutes avant d'essayer de vous connecter et que vous ne parvenez toujours pas à vous connecter, il se peut que vous deviez appliquer manuellement la configuration de gestion des hôtes à votre instance. Utilisez la procédure suivante pour mettre à jour une configuration de gestion des hôtes Quick Setup et appliquer des modifications à une instance.

Pour mettre à jour une configuration de gestion d'hôtes en utilisant Quick Setup

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Quick Setup.
3. Dans la liste Configurations, choisissez la configuration de Gestion des hôtes que vous avez créée.
4. Choisissez Actions, puis Enregistrer la configuration.
5. Dans la section Cibles, choisissez Manuel.
6. Dans la section Instances, choisissez l'instance que vous avez créée.
7. Choisissez Mettre à jour.

Attendez quelques minutes pour qu'EC2 actualise l'onglet Session Manager. Si vous ne parvenez toujours pas à vous connecter ou si vous recevez un message d'erreur, consultez les autres solutions à ce problème.

Solution C : erreur concernant l'absence de l'SSM Agent

Si vous n'avez pas pu créer une configuration de gestion des hôtes en utilisant Quick Setup, ou si vous recevez un message d'erreur indiquant que l'SSM Agent n'est pas installé, il se peut que vous deviez installer manuellement l'SSM Agent sur votre instance. SSM Agent est un logiciel Amazon qui permet à Systems Manager de se connecter à votre instance en utilisant Session Manager. L'SSM Agent est installé par défaut sur la plupart des Amazon Machine Images (AMI). Si votre instance a été créée à partir d'une AMI non standard ou d'une ancienne AMI, vous devrez peut-être installer l'agent manuellement. Pour la procédure d'installation de SSM Agent, consultez la rubrique suivante qui correspond au système d'exploitation de votre instance.

- Windows Server
- macOS
- AlmaLinux
- Amazon Linux 1
- Amazon Linux 2 et AL2023
- CentOS
- CentOS Stream
- Debian Server
- Oracle Linux
- Red Hat Enterprise Linux
- Rocky Linux
- SUSE Linux Enterprise Server
- Ubuntu Server

Pour les problèmes liés à SSM Agent, consultez Résolution des problèmes de SSM Agent.

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 : aucun administrateur système ne vous a accordé AWS Identity and Access Management (IAM) d'autorisations de politique pour démarrer des Session Manager sessions. Pour obtenir des informations, veuillez consulter Contrôler les accès de session utilisateur aux instances.

Je n'ai pas le droit de modifier les préférences de session

Problème : vous essayez de mettre à jour des 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 politique IAM pour configurer des préférences Session Manager. Pour plus d'informations, veuillez consulter Accorder ou révoquer des autorisations utilisateur pour mettre à jour des préférences Session Manager.

Nœud géré non disponible ou non configuré pour Session Manager

Problème 1 : vous souhaitez démarrer une session sur la page de la console Start a session (Démarrer une session), mais le nœud géré ne figure pas dans la liste.

- Solution A : Le nœud géré auquel vous souhaitez vous connecter n'a peut-être pas été configuré AWS Systems Manager. Pour plus d'informations, consultez Con AWS Systems Manager figuration.

  > **Note**
  >
  > S'il AWS Systems Manager SSM Agent est déjà exécuté sur un nœud géré lorsque vous attachez le profil d'instance IAM, vous devrez peut-être redémarrer l'agent avant que l'instance ne soit répertoriée sur la page de console Démarrer une session.

- Solution B : la configuration de proxy que vous avez appliquée à l'SSM Agent sur votre nœud géré peut être incorrecte. Si la configuration du proxy est incorrecte, le nœud géré ne sera pas en mesure d'atteindre les points de terminaison de service nécessaires, ou le nœud pourra signaler un système d'exploitation différent à Systems Manager. Pour plus d'informations, consultez Configuration SSM Agent pour utiliser un proxy sur les nœuds Linux et Configurer l'SSM Agent pour utiliser un proxy pour les instances Windows Server.

Problème 2 : un nœud géré auquel vous souhaitez vous connecter figure dans la liste sur la page de la console Start a session (Démarrer une session), mais la page indique que « L'instance que vous avez sélectionnée n'est pas configurée pour utiliser Session Manager ».

- Solution A : le nœud géré a été configuré pour une utilisation avec le service Systems Manager, mais le profil d'instance IAM attaché au nœud n'inclut peut-être pas les autorisations relatives à la

fonctionnalité Session Manager. Pour en savoir plus, consultez la page Vérifier ou créer un profil d'instance IAM avec des autorisations Session Manager.

- Solution B : le nœud géré n'exécute pas une version de l'SSM Agent qui prend en charge Session Manager. Mettez à jour l'SSM Agent du nœud vers la version 2.3.68.0 ou ultérieure.

  Mettez manuellement à jour l'SSM Agent sur un nœud géré en suivant la procédure décrite sur les pages Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Windows Server, Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Linux ou Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour macOS, selon votre système d'exploitation.

  Vous pouvez également utiliser le document Run Command `AWS-UpdateSSMAgent` pour mettre à jour la version de l'agent sur un ou plusieurs nœuds gérés à la fois. Pour plus d'informations, veuillez consulter Mise à jour de SSM Agent à l'aide de Run Command.

  > **Tip**
  >
  > Pour garder constamment votre agent à jour, nous vous recommandons de configurer la mise à jour automatique de l'SSM Agent vers la dernière version, tel qu'expliqué ci-dessous :
  >
  > - Exécutez `AWS-UpdateSSMAgent` dans le cadre d'une association State Manager. Pour plus d'informations, veuillez consulter Démonstration : Mise à jour automatique de l'SSM Agent (CLI).
  > - 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) et Didacticiel : Créer et configurer une fenêtre de maintenance (AWS CLI).

- Solution C : le nœud géré ne peut pas atteindre les points de terminaison de service nécessaires. Vous pouvez améliorer le niveau de sécurité de vos nœuds gérés en utilisant des points de terminaison d'interface optimisés AWS PrivateLink pour vous connecter aux points de terminaison Systems Manager. L'alternative à l'utilisation de points de terminaison d'interface consiste à activer l'accès Internet sortant sur vos nœuds gérés. Pour plus d'informations, consultez Utiliser PrivateLink pour configurer un point de terminaison VPC pour. Session Manager
- Solution D : le nœud géré dispose de ressources d'UC ou de mémoire limitées. Même si votre nœud géré est fonctionnel, s'il 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.

Plugin Session Manager introuvable

Pour utiliser les commandes AWS CLI d'exécution de session, le Session Manager plugin doit également être installé sur votre machine locale. Pour plus d'informations, veuillez consulter Installez le Session Manager plugin pour AWS CLI.

Plug-in Session Manager pas ajouté automatiquement au chemin de la ligne de commande (Windows)

Lorsque vous installez le plug-in Session Manager 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 plugin Session Manager 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 saisissez **`environment variables`**.
2. Sélectionnez Modifier les variables d'environnement pour votre compte.
3. Sélectionnez PATH, puis Modifier.
4. Ajoutez des chemins d'accès dans le champ Variable value (Valeur de la variable), en les séparant par des points virgules, tel qu'illustré dans l'exemple : *`C:\existing\path`*;***`C:\new\path`***

   `C:\existing\path` représente la valeur déjà dans le champ. ***`C:\new\path`*** représente le chemin que vous souhaitez ajouter, comme indiqué dans ces exemples.

   - machines 64 bits : `C:\Program Files\Amazon\SessionManagerPlugin\bin\`
   - machines 32 bits : `C:\Program Files (x86)\Amazon\SessionManagerPlugin\bin\`
5. Sélectionnez OK deux fois pour appliquer les nouveaux paramètres.
6. Fermez toute invite de commande en cours d'exécution et rouvrez.

Le plugin Session Manager ne répond pas

Durant une session de réacheminement de port, si un logiciel antivirus est installé sur votre ordinateur local, le réacheminement du trafic peut s'arrêter. Dans certains cas, l'interférence d'un logiciel antivirus avec le plugin Session Manager provoque des deadlocks. Pour résoudre ce problème, autorisez ou excluez le plugin Session Manager dans le logiciel antivirus. Pour obtenir des informations sur le chemin d'installation par défaut pour le plugin Session Manager, veuillez consulter Installez le Session Manager plugin pour AWS CLI.

TargetNotConnecté

Problème : vous essayez de démarrer une session, mais le système renvoie le message d'erreur suivant : « Une erreur s'est produite (TargetNotConnected) lors de l'appel de l' StartSession opération : *`InstanceID`* n'est pas connecté ».

- Solution A : cette erreur est renvoyée lorsque le nœud géré cible spécifié pour la session n'est pas entièrement configuré pour être utilisé avec Session Manager. Pour plus d'informations, veuillez consulter Configuration de Session Manager.
- Solution B : Cette erreur est également renvoyée si vous tentez de démarrer une session sur un nœud géré situé dans un autre Compte AWS ou Région AWS.

Affichage d'un écran vide après le démarrage d'une session

Problème: vous démarrez une session et Session Manager affiche un écran vide.

- Solution A : ce problème peut se produire lorsque le volume racine du nœud géré est plein. En raison du manque d'espace disque, l'SSM Agent sur le nœud géré 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 la section Collecter des métriques, des journaux et des traces avec l' CloudWatch agent dans le guide de CloudWatch l'utilisateur Amazon.
- 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 Région AWS spécifiée :

```
https://us-west-2.console.aws.amazon.com/systems-manager/session-manager/sessions?
region=us-west-1
```

- Solution C : Le nœud géré se connecte à Systems Manager via des points de terminaison VPC, et vos Session Manager préférences écrivent le résultat de la session dans un compartiment Amazon S3 ou un groupe de CloudWatch journaux Amazon Logs, mais aucun point de terminaison de `s3` passerelle ou d'`logs`interface n'existe dans le VPC. Un point de terminaison `s3` au format **`com.amazonaws.`*`region`*`.s3`** est requis si vos nœuds gérés se connectent à Systems Manager à l'aide de points de terminaison de VPC et que vos préférences Session Manager écrivent la sortie de session dans un compartiment Amazon S3. Un `logs` point de terminaison au format **`com.amazonaws.`*`region`*`.logs`**est également requis si vos nœuds gérés se connectent à Systems Manager via des points de terminaison VPC et si vos Session Manager préférences écrivent le résultat de la session dans un groupe de CloudWatch journaux Logs. Pour plus d'informations, consultez Création de points de terminaison de VPC pour Systems Manager.
- Solution D : le groupe de journaux ou le compartiment Amazon 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 Amazon S3 que vous avez spécifié dans vos préférences de session n'est pas chiffré, mais vous avez défini l'entrée `cloudWatchEncryptionEnabled` ou `s3EncryptionEnabled` sur `true`. Pour résoudre ce problème, mettez à jour vos préférences de session avec un groupe de journaux ou un compartiment Amazon S3 chiffré, ou définissez l'entrée `cloudWatchEncryptionEnabled` ou `s3EncryptionEnabled` sur `false`. Ce scénario s'applique uniquement aux clients qui créent des préférences de session avec les outils de ligne de commande.

Le nœud géré cesse de répondre lorsque les sessions durent longtemps

Problème : votre nœud géré ne répond plus ou se bloque lorsqu'une session dure longtemps.

Solution : réduisez la durée de conservation des journaux de l'SSM Agent pour Session Manager.

Pour réduire la durée de conservation des journaux de l'SSM Agent pour les sessions

1. Localisez le `amazon-ssm-agent.json.template` dans le répertoire `/etc/amazon/ssm/` pour Linux, ou `C:\Program Files\Amazon\SSM` pour Windows.
2. Copiez le contenu du `amazon-ssm-agent.json.template` dans un nouveau fichier nommé `amazon-ssm-agent.json`, dans le même répertoire.
3. Réduisez la valeur par défaut de la valeur `SessionLogsRetentionDurationHours` dans la propriété `SSM` et enregistrez le fichier.

4. Redémarrez SSM Agent.

Une erreur s'est produite (InvalidDocument) lors de l'appel de l' StartSession opération

Problème : Le message d'erreur suivant s'affiche lorsque vous démarrez une session à l'aide de AWS CLI.

```
An error occurred (InvalidDocument) when calling the StartSession operation: Document
 type: 'Command' is not supported. Only type: 'Session' is supported for Session
 Manager.
```

Solution : Le document SSM que vous avez spécifié pour le paramètre `--document-name` n'est pas un document de Session. Utilisez la procédure suivante pour afficher la liste des documents de session dans la AWS Management Console.

Pour afficher la liste des documents de session

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.
3. Dans la liste des Catégories, sélectionnez Documents de session.

AWS Systems Manager Run Command

En utilisant Run Command une fonctionnalité de AWS Systems Manager, vous pouvez gérer à distance et en toute sécurité la configuration de vos nœuds gérés. Un nœud géré est une instance Amazon Elastic Compute Cloud (Amazon EC2) ou une machine non EC2 de votre environnement hybride et multi-cloud qui a été configurée pour Systems Manager. Run Command vous permet d'automatiser les tâches administratives courantes et d'effectuer des modifications de configuration ponctuelles à grande échelle. Vous pouvez utiliser Run Command à partir du AWS Management Console, du AWS Command Line Interface (AWS CLI) ou des AWS SDK. AWS Tools for Windows PowerShellRun Commandest offert sans frais supplémentaires. Pour vos premiers pas dans Run Command, ouvrez Systems Manager console. Dans le panneau de navigation, sélectionnez Run Command.

Les administrateurs ont recours à la fonctionnalité Run Command pour 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, joindre des instances à un domaine Windows et bien plus.

Démarrage

Le tableau suivant comporte des informations pour vous aider à vous familiariser avec Run Command.

Rubrique	Détails
Con AWS Systems Manager figuration	Vérifiez que vous avez satisfait la configura tion requise pour vos instances Amazon Elastic Compute Cloud (Amazon EC2) et vos machines non EC2 dans un environnement hybride et multicloud.
Utilisation de Systems Manager dans des environnements hybrides et multicloud	(Facultatif) Enregistrez des serveurs et des machines virtuelles sur site AWS afin de pouvoir les gérer à l'aide de. Run Command
the section called “Gestion des appareils de pointe avec Systems Manager”	(Facultatif) Configurez les appareils de périphérie pour pouvoir les gérer à l'aide de Run Command.
Exécution de commandes sur des nœuds gérés	Découvrez comment exécuter une commande ciblant un ou plusieurs nœuds gérés à l'aide de la AWS Management Console.
Procédures Run Command	Apprenez à exécuter des commandes à l'aide des outils pour Windows PowerShell ou du AWS CLI.

EventBridge soutien

Cette fonctionnalité de Systems Manager est prise en charge à la fois en tant que type d'événement et en tant que type de cible dans EventBridge les règles Amazon. Pour plus d'informations, consultez Surveillance d'événements Systems Manager avec Amazon EventBridge et Référence : modèles et types d'événements Amazon EventBridge pour Systems Manager.

Plus d'informations

- Run Command à distance sur une instance EC2 (didacticiel de 10 minutes)
- Service Quotas Systems Manager de la Référence générale d'Amazon Web Services
- AWS Systems Manager API Reference

Rubriques

Configuration de Run Command

Avant de pouvoir gérer des nœuds à l'aide de Run Command (fonction développée par AWS Systems Manager), vous devez configurer une politique AWS Identity and Access Management (IAM) pour tout utilisateur qui exécutera des commandes.

Vous devez également configurer vos nœuds pour Systems Manager. Pour de plus amples informations, veuillez consulter Con AWS Systems Manager figuration.

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 nœuds gérés.

Surveiller les exécutions de commandes avec Amazon EventBridge

Vous pouvez utiliser 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 Run Command comme action cible quand un événement EventBridge a lieu. Pour de plus amples informations, veuillez consulter Configurer EventBridge pour des événements Systems Manager.

Surveiller les exécutions de commandes avec Amazon CloudWatch Logs

Vous pouvez configurer Run Command pour envoyer régulièrement tous les journaux de sortie de commande et d'erreurs à un groupe de journaux Amazon CloudWatch. 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 de plus amples informations, veuillez consulter Configuration d'Amazon CloudWatch Logs pour Run Command.

Restriction de l'accès Run Command à des nœuds gérés spécifiques

Vous pouvez restreindre la capacité d'un utilisateur à exécuter des commandes sur des nœuds gérés en utilisant AWS Identity and Access Management (IAM). Plus précisément, vous pouvez créer une politique IAM avec une condition selon laquelle l'utilisateur ne peut exécuter de commandes que sur les nœuds gérés présentant des balises spécifiques. Pour de plus amples informations, veuillez consulter Restriction de l'accès Run Command en fonction des balises.

Restriction de l'accès Run Command en fonction des balises

Cette section explique comment restreindre la capacité d'un utilisateur à exécuter des commandes sur des nœuds gérés en spécifiant une condition de balise dans une politique IAM. Les nœuds gérés incluent les instances Amazon EC2 et les nœuds non EC2 d'un environnement hybride et multicloud configurés pour Systems Manager. Bien que les informations ne soient pas explicitement présentées, vous pouvez également restreindre l'accès aux appareils Core AWS IoT Greengrass gérés. Pour commencer, vous devez baliser vos appareils AWS IoT Greengrass. Pour plus d'informations, consultez Baliser vos ressources AWS IoT Greengrass Version 2 dans le Guide du développeur AWS IoT Greengrass Version 2.

Vous pouvez restreindre l'exécution de commandes à des nœuds gérés spécifiques en créant une politique IAM qui comporte une condition selon laquelle l'utilisateur ne peut exécuter de commandes que sur les nœuds comportant des balises spécifiques. Dans l'exemple suivant, l'utilisateur est autorisé à utiliser Run Command (`Effect: Allow, Action: ssm:SendCommand`) en utilisant n'importe quel document SSM (`Resource: arn:aws:ssm:*:*:document/*`) sur n'importe quelle instance (`Resource: arn:aws:ec2:*:*:instance/*`) à condition que le nœud soit un serveur Web Finance (`ssm:resourceTag/Finance: WebServer`). Si l'utilisateur envoie une commande à un nœud non balisé 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 politiques IAM qui permettent à un utilisateur d'exécuter des commandes sur des nœuds gérés balisés à l'aide de plusieurs balises. La politique suivante permet à l'utilisateur d'exécuter des commandes sur des nœuds gérés dotés de deux balises. Si un utilisateur envoie une commande à un nœud non balisé 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 politiques IAM qui permettent à un utilisateur d'exécuter des commandes sur plusieurs groupes de nœuds gérés balisés. L'exemple de politique suivante permet à l'utilisateur d'exécuter des commandes sur un des deux groupes de nœuds balisés, 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 plus d'informations sur la création de politiques IAM, consultez Politiques gérées et politiques en ligne dans le Guide de l'utilisateur IAM. Pour plus d'informations sur le balisage des nœuds gérés, consultez Tag Editor dans le Guide de l'utilisateur AWS Resource Groups.

Exécution de commandes sur des nœuds gérés

Cette section comprend des informations sur le mode d'envoi de commandes depuis la console AWS Systems Manager vers des nœuds gérés. Cette section inclut également des informations sur l'annulation d'une commande.

Pour de plus amples informations sur l'envoi de commandes à l'aide de Windows PowerShell, consultez Procédure pas à pas : utilisez le AWS Tools for Windows PowerShell avec Run Command ou les exemples de la section AWS Systems Manager de la Référence AWS Tools for PowerShell Cmdlet. Pour de plus amples informations sur l'envoi de commandes à l'aide de l'AWS Command Line Interface (AWS CLI), consultez Procédure : utiliser les AWS CLI avec Run Command dans la Référence de la CLI SSM.

> **Important**
>
> Lorsque vous envoyez une commande à l'aide de Run Command, n'incluez pas d'informations sensibles formatées en texte brut, comme des mots de passe, des données de configuration ou d'autres secrets. L'activité de l'API Systems Manager utilisée dans votre compte est consignée dans un compartiment S3 pour les journaux AWS CloudTrail. Cela signifie que tout utilisateur ayant accès à ce compartiment S3 peut consulter les valeurs en texte brut de ces secrets. Pour cette raison, nous vous recommandons de créer et d'utiliser des paramètres `SecureString` pour chiffrer les données sensibles que vous utilisez dans le cadre de vos opérations Systems Manager.
> Pour de plus amples informations, veuillez consulter Restriction de l'accès aux paramètres Systems Manager à l'aide des politiques IAM.

Table des matières

- Annulation d'une commande

Exécution des commande à partir de la console

Vous pouvez utiliser Run Command une fonctionnalité de AWS Systems Manager, from AWS Management Console pour configurer des nœuds gérés sans avoir à vous y connecter. Cette rubrique comprend un exemple qui montre comment mettre à jour SSM Agent sur un nœud géré à l'aide de la fonctionnalité Run Command.

Avant de commencer

Avant d'envoyer une commande avec Run Command, vérifiez que vos nœuds gérés respectent la configuration requise de Systems Manager.

Pour envoyer une commande à l'aide de la fonctionnalité Run Command

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Run Command.
3. Sélectionnez Run Command (Exécuter la commande).
4. Dans la liste Command document (Document de commande), sélectionnez 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 Targets (Cibles), sélectionnez les nœuds gérés sur lesquels vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant des instances ou des appareils de périphérie manuellement ou en spécifiant un groupe de ressources.

   > **Tip**
   >
   > Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés 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. Pour Rate control (Contrôle de débit) :

   - Dans Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage de nœuds gérés sur lesquels exécuter simultanément la commande.

     > **Note**
     >
     > Si vous avez sélectionné des cibles en spécifiant les balises appliquées aux nœuds gérés ou en spécifiant des groupes de ressources AWS , et que vous n'êtes pas certain du nombre de nœuds gérés ciblés, limitez le nombre de cibles autorisées à exécuter simultanément le document en indiquant un pourcentage.

   - Dans Error threshold (Seuil d'erreur), indiquez quand arrêter l'exécution de la commande sur les autres nœuds gérés après l'échec de celle-ci sur un certain nombre ou un certain pourcentage de nœuds. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les nœuds gérés sur lesquels la commande est toujours en cours de traitement peuvent également envoyer des erreurs.

9. (Facultatif) Choisissez une CloudWatch alarme à appliquer à votre commande de surveillance. Pour associer une CloudWatch alarme à votre commande, le principal IAM qui exécute la commande doit être autorisé à effectuer l'`iam:createServiceLinkedRole`action. Pour plus d'informations sur les CloudWatch alarmes, consultez la section Utilisation des CloudWatch alarmes Amazon. Notez que l'activation de votre alarme empêche l'exécution des appels de commande en attente.

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). Saisissez 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 (pour les instances EC2) ou de la fonction du service IAM (pour les machines activées par un système hybride) attribués à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems

Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, assurez-vous que le profil d'instance ou le rôle de service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

11. Dans la section SNS notifications (Notifications SNS), si vous souhaitez envoyer des notifications sur le statut d'exécution des commandes, cochez la case Enable SNS notifications (Activer les notifications SNS).

    Pour plus d'informations sur la configuration des notifications Amazon SNS pour Run Command, consultez Surveillance des changements d'état du Systems Manager à l'aide des notifications Amazon SNS.

12. Cliquez sur Exécuter.

Pour plus d'informations sur l'annulation d'une commande, consultez the section called "Annulation d'une commande".

Réexécution des commandes

Systems Manager comprend deux options pour vous aider à réexécuter une commande à partir de la page Run Command (Exécuter la commande) de la console Systems Manager.

- Réexécuter : ce bouton vous permet d'exécuter la même commande sans y apporter de modifications.
- Copier vers 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 AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Run Command.
3. Sélectionnez 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 Command history (Historique des commandes).

4. Sélectionnez Réexécuter pour exécuter la même commande sans modifications, ou sélectionnez Copier vers nouveau pour modifier les paramètres de la commande avant de l'exécuter.

Exécution de commandes à l'aide d'une version de document spécifique

Vous pouvez utiliser le paramètre document-version pour spécifier la version d'un document AWS Systems Manager à utiliser lors de l'exécution de la commande. Vous pouvez spécifier l'une des options suivantes pour ce paramètre :

- $DEFAULT
- $LATEST
- Version number

Appliquez 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 Command Line Interface (AWS CLI).

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l'AWS CLI.

2. Répertorier tous les documents disponibles

   Cette commande répertorie tous les documents disponibles pour votre compte en fonction des autorisations AWS Identity and Access Management (IAM).

   ```
   aws ssm list-documents
   ```

3. Utilisez la commande suivante pour afficher les différentes versions d'un document. Remplacez *document name* (nom du document) avec vos propres informations.

   ```
   aws ssm list-document-versions \
       --name "document name"
   ```

4. Exécutez la commande suivante pour exécuter une commande qui utilise une version de document SSM. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

```
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 Command Line Interface (AWS CLI).

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l'AWS CLI.

2. Répertorier tous les documents disponibles

   Cette commande répertorie tous les documents disponibles pour votre compte en fonction des autorisations AWS Identity and Access Management (IAM).

```
aws ssm list-documents
```

3. Utilisez la commande suivante pour afficher les différentes versions d'un document. Remplacez *document name* (nom du document) avec vos propres informations.

```
aws ssm list-document-versions ^
    --name "document name"
```

4. Exécutez la commande suivante pour exécuter une commande qui utilise une version de document SSM. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

```
aws ssm send-command ^
    --document-name "AWS-RunShellScript" ^
    --parameters commands="echo Hello" ^
```

```
    --instance-ids instance-ID ^
    --document-version "$LATEST"
```

PowerShell

Pour exécuter des commandes avec les Tools for PowerShell

1. Si vous ne l'avez pas déjà fait, installez et configurez AWS Tools for PowerShell (outils pour Windows PowerShell).

   Pour plus d'informations, consultez Installation d'AWS Tools for PowerShell.
2. Répertorier tous les documents disponibles

   Cette commande répertorie tous les documents disponibles pour votre compte en fonction des autorisations AWS Identity and Access Management (IAM).

   ```
   Get-SSMDocumentList
   ```
3. Utilisez la commande suivante pour afficher les différentes versions d'un document. Remplacez *document name* (nom du document) avec vos propres informations.

   ```
   Get-SSMDocumentVersionList `
       -Name "document name"
   ```
4. Exécutez la commande suivante pour exécuter une commande qui utilise une version de document SSM. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   ```
   Send-SSMCommand `
       -DocumentName "AWS-RunShellScript" `
       -Parameter @{commands = "echo helloWorld"} `
       -InstanceIds "instance-ID" `
       -DocumentVersion $LATEST
   ```

Exécuter des commandes à grande échelle

Vous pouvez utiliser Run Command, une fonctionnalité d'AWS Systems Manager, pour exécuter des commandes sur une flotte de nœuds gérés à l'aide des `targets`. Le paramètre `targets` accepte une combinaison `Key,Value` basée sur les balises que vous avez spécifiées pour vos

nœuds gérés. Lorsque vous exécutez la commande, le système recherche les fichiers et tente d'exécuter la commande sur tous les nœuds gérés qui correspondent aux balises spécifiées. Pour plus d'informations sur le balisage des instances gérées, veuillez consulter la rubrique Balisage de vos ressources AWS dans le Guide de l'utilisateur du balisage des ressources AWS. Pour plus d'informations sur le balisage des appareils IoT gérés, consultez.Baliser vos ressources AWS IoT Greengrass Version 2 dans le Guide du développeur AWS IoT Greengrass Version 2.

Vous pouvez aussi utiliser le paramètre `targets` pour cibler une liste d'ID de nœud géré spécifiques, tel que décrit dans la section suivante.

Pour contrôler l'exécution d'une commande sur des centaines de milliers de nœuds gérés, la fonctionnalité Run Command inclut également des paramètres pour limiter le nombre de nœuds gérés 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 nœuds gérés

Vous pouvez exécuter une commande et cibler des nœuds gérés en spécifiant des balises, des noms de groupes de ressources AWS ou des ID de nœud géré.

Les exemples suivants montrent le format de commande lorsque vous utilisez Run Command depuis la AWS Command Line Interface (AWS CLI). Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations. Les exemples de commandes de cette section sont tronqués à l'aide de `[...]`.

Exemple 1 : ciblage de balises

Linux & macOS

```
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 AWS Identity and Access Management (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 AWS Resource Groups.

Linux & macOS

```
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 au maximum cinq types 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 AWS Resource Groups.

Linux & macOS

```
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 des ID d'instance

Les exemples suivants montrent comment cibler les nœuds gérés à l'aide de la clé `instanceids` avec le paramètre `targets`. Vous pouvez utiliser cette clé pour cibler les appareils Core AWS IoT Greengrass gérés car chaque appareil se voit affecter un mi-*ID_number*. Vous pouvez afficher les ID d'appareil dans Fleet Manager, une fonctionnalité de AWS Systems Manager.

Linux & macOS

```
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 nœuds gérés pour différents environnements à l'aide d'une `Key` nommée `Environment` et des `Values` de `Development`, `Test`, `Pre-production` et `Production`, vous pourrez donc envoyer une commande à tous les nœuds gérés dans l'un de ces environnements à l'aide du paramètre `targets` avec la syntaxe suivante.

Linux & macOS

```
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 nœuds gérés supplémentaires dans d'autres environnements en ajoutant un élément à la liste `Values`. Séparez les éléments avec des virgules.

Linux & macOS

```
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 : affiner vos cibles à l'aide de plusieurs critères `Key`

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 nœuds gérés qui répondent à tous les critères. La commande suivante cible tous les nœuds gérés balisés pour le service financier et balisés pour le rôle de serveur de base de données.

Linux & macOS

```
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 critères `Key` et `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 & macOS

```
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 de nœuds gérés balisés à l'aide de plusieurs critères `Values`

Si vous avez balisé des nœuds gérés pour différents environnements à l'aide d'une `Key` nommée `Department` et `Values` de `Sales` et `Finance`, vous pouvez envoyer une commande à tous les nœuds gérés dans l'un de ces environnements à l'aide du paramètre `targets` avec la syntaxe suivante.

Linux & macOS

```
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 ^
    [...]
```

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, placez la clé ou la valeur de balise entre guillemets, comme illustré dans les exemples suivants.

Exemple : espaces dans la balise `Value`

Linux & macOS

```
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 : espaces dans la clé `tag` et dans `Value`

Linux & macOS

```
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 : espaces dans un élément d'une liste de `Values`

Linux & macOS

```
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 nœuds gérés d'un groupe à l'aide des contrôles de concurrence et des contrôles d'erreur.

Rubriques

- Utilisation de contrôles d'accès simultanés
- Utilisation de contrôles d'erreur

Utilisation de contrôles d'accès simultanés

Le contrôle de nombre de nœuds gérés exécutant une commande simultanément est possible à l'aide du `max-concurrency` paramètre des options (Simultanéité de la page Exécuter une commande). Vous pouvez spécifier un nombre absolu de nœuds géré, 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 un seul nœud et attend jusqu'à ce que le système reconnaisse l'appel initial avant d'envoyer la commande à deux autres nœuds. Le système envoie de façon exponentielle des commandes à plusieurs nœuds 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 & macOS

```
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 de nœuds gérés 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 nœuds gérés. Vous pouvez spécifier un nombre absolu d'erreurs (par exemple, **10**) ou un pourcentage de l'ensemble de la cible (par exemple, **10%**). Si, par exemple, 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 nœuds gérés supplémentaires une fois que le premier résultat d'erreur est renvoyé. Si vous envoyez une commande à 50 nœuds gérés et que vous définissez `max-errors` avec la valeur **10%**, le système arrête d'envoyer la commande aux nœuds gérés supplémentaires à la réception de la sixième erreur.

Les appels qui exécutent déjà une commande lorsque `max-errors` est atteint sont autorisés à se terminer, mais certains de ces appels pourraient 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 & macOS

```
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 Pending (En attente) ou Executing (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 annulée et que le processus sous-jacent sera arrêté.

Pour annuler une commande via la console

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Run Command.
3. Sélectionnez l'appel de commande que vous souhaitez annuler.
4. Sélectionnez Annuler la commande.

Pour annuler une commande à l'aide du AWS CLI

Exécutez la commande suivante. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

Linux & macOS

```
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.

Utilisation des codes de sortie dans les commandes

Dans certains cas, vous devrez peut-être gérer la façon dont vos commandes sont gérées à l'aide de codes de sortie.

Spécifier les codes de sortie dans les commandes

À l'aide de Run Command, une fonctionnalité d'AWS Systems Manager, vous pouvez spécifier des codes de sortie pour déterminer la manière dont les commandes sont gérées. 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 du script lors du premier échec de la commande, 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
```

Scripts PowerShell

PowerShell exige que vous appeliez explicitement `exit` dans vos scripts pour que Run Command capture 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
```

Gestion des redémarrages lors de l'exécution de commandes

Si vous utilisez Run Command une fonctionnalité de AWS Systems Manager, pour exécuter des scripts qui redémarrent les nœuds gérés, nous vous recommandons de spécifier un code de sortie dans votre script. Si vous essayez de redémarrer un nœud 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 nœuds gérés Windows, spécifiez `exit 3010` dans votre script. Pour les nœuds gérés Linux et macOS, spécifiez `exit 194`. Le code de sortie indique à AWS Systems Manager l'agent (SSM Agent) de redémarrer le nœud géré, puis de redémarrer le script une fois le redémarrage terminé. Avant de commencer le redémarrage, SSM Agent informe le service Systems Manager dans le cloud que les communications seront interrompues pendant le redémarrage du serveur.

> **Note**
>
> Le script de redémarrage ne peut pas faire partie d'un plugin `aws:runDocument`. Si un document contient le script de redémarrage et qu'un autre document tente d'exécuter ce document via le plugin `aws:runDocument`, SSM Agent renvoie une erreur.

Création de scripts idempotents

Lorsque vous développez des scripts qui redémarrent les nœuds gérés, 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 le nœud géré 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
    }
```

Exemples

Les échantillons de script suivants utilisent des codes de sortie pour redémarrer les nœuds gérés. L'exemple sur Linux installe les mises à jour de package sur Amazon Linux, puis redémarre le nœud. L'Windows Serverexemple installe le client Telnet sur le nœud, puis le redémarre.

Amazon Linux

```
#!/bin/bash
yum -y update
needs-restarting -r
if [ $? -eq 1 ]
then
        exit 194
```

```
else
        exit 0
fi
```

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
    }
```

Comprendre les états des commandes

Run Command, une fonctionnalité de AWS Systems Manager, fournit des informations d'état détaillées sur les différents états rencontrés par une commande pendant le traitement et pour chaque nœud géré qui a traité la commande. Vous pouvez surveiller les statuts de commande à l'aide des méthodes suivantes :

- Cliquez sur l'icône Refresh (Actualiser) dans l'onglet Commands (Commandes) de l'interface de la console Run Command.
- Appelez les commandes de liste ou les appels de commandes de liste à l'aide du (). AWS Command Line Interface AWS CLIOu appelez Get-SSMCommand ou Get-SSM en utilisant. CommandInvocation AWS Tools for Windows PowerShell
- Configurez Amazon EventBridge pour qu'il réponde à un état ou à un changement de statut.
- Configurez Amazon Simple Notification Service (Amazon SNS) de sorte à envoyer des notifications pour tous les changements de statut ou pour des statuts spécifiques, comme `Failed` ou `TimedOut`.

État Run Command

La fonctionnalité Run Command 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 SSM de votre commande. Pour de plus amples informations sur les plug-ins, consultez Référence de plug-in de document Command.

Lorsque vous envoyez une commande à plusieurs nœuds gérés en même temps, chaque copie de la commande qui cible chaque nœud correspond à une invocation de commande. Par exemple, si vous utilisez le document `AWS-RunShellScript` et que vous envoyez une commande `ifconfig` à 20 instances Linux, cette commande aura 20 appels. Chaque invocation de commande signale le statut individuellement. Les plug-ins d'une invocation de commande donné communiquent aussi le statut individuellement.

Enfin, la fonctionnalité Run Command 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 de nœuds gérés 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 Exécuter des commandes à grande échelle.

Statut détaillé pour des plug-ins et des appels de commande

Statut	Détails
En attente	La commande n'a pas encore été envoyée au nœud géré ou n'a pas été reçue par l'SSM Agent. Si la commande n'est pas reçue par l'agent avant le délai prévu, qui est égal à la somme du paramètre Timeout (seconds) (Délai d'expiration (secondes)) et le paramètre Execution timeout (Délai d'exécution), le statut passe à `Delivery Timed Out`.
InProgress	Systems Manager tente d'envoyer la commande au nœud géré, ou la commande a été reçue par l'SSM Agent et a commencé à s'exécuter sur l'instance. Selon le résultat de tous les plug-ins de commande, le statut passe à `Success`, `Failed`, `Delivery Timed`

Statut	Détails
	`Out` ou `Execution Timed Out`. Exception : si l'agent n'est pas en cours d'exécution ou n'est pas disponible sur le nœud, le statut de la commande reste sur `In Progress` jusqu'à ce que l'agent soit à nouveau disponible ou que la limite du délai d'exécution soit atteinte. Le statut est ensuite remplacé par un état de mise hors service.
Delayed (Retardé)	Le système tente d'envoyer la commande au nœud géré, mais n'a pas réussi. Le système réessaie.

Statut	Détails
Réussite	Ce statut est renvoyé sous diverses condition s. Ce statut ne signifie pas que la commande a été effectuée sur le nœud. Par exemple, la commande peut être reçue par SSM Agent le nœud géré et renvoyer un code de sortie égal à zéro si vous PowerShell `ExecutionPolicy` empêchez l'exécution de la commande. Il s'agit d'un statut de terminal. Les conditions qui entraînent le renvoi d'un `Success` statut par une commande sont les suivantes : • Lors du ciblage d'une instance unique, la commande a été reçue par SSM Agent le nœud géré et a renvoyé un code de sortie égal à zéro. • Lorsque vous ciblez plusieurs instances, le nombre d'appels ayant échoué n'a pas dépassé le seuil d'erreur spécifié dans la commande. • Lorsque vous ciblez plusieurs instances, au moins une invocation a réussi tandis que les autres ont expiré. Le seuil d'erreur spécifié s'applique toujours. • Lorsque vous ciblez une balise, aucune instance associée à la balise n'est trouvée. • Lorsque vous ciblez une balise, le nombre d'appels ayant échoué n'a pas dépassé le seuil d'erreur spécifié dans la commande. • Lorsque vous ciblez un tag, au moins un appel a réussi alors que les autres ont expiré. Le seuil d'erreur spécifié s'applique toujours. • Des applications ou des politiques appliquée s au niveau du système d'exploitation empêchent ou annulent l'exécution d'une

Statut	Détails
	commande, ce qui entraîne le renvoi d'un code de sortie égal à zéro. **Note** Les mêmes conditions s'appliquent lorsque vous ciblez des groupes de ressources. 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).
DeliveryTimedDehors	La commande n'a pas été fournie au nœud géré avant l'expiration du délai total. Les dépassements de délai totaux ne sont pas comptabilisés dans la limite `max-errors` de la commande parent, mais ils permettent de savoir si l'état de la commande parent est `Success`, `Incomplete` ou `Delivery Timed Out`. Il s'agit d'un statut de terminal.
ExecutionTimedDehors	L'automatisation de la commande a démarré sur le nœud géré, mais l'exécution de la commande ne s'est pas terminée avant l'expiration du délai d'exécution. Les expirations de délais d'exécution sont considérés comme des échecs, ce qui enverra une réponse nulle et Systems Manager quittera sortira de la tentative d'exécution de l'automatisation des commandes et signalera échec comme état.

Statut	Détails
Échec	La commande n'a pas réussi sur le nœud géré. Pour un plugin, cela signifie que le code de résultat n'était pas zéro. Pour une invocation de commande, cela signifie que le code de résultat pour un ou plusieurs plugins n'était pas zéro. Les échecs d'invocation sont comptabilisés dans la limite `max-errors` de la commande parent. Il s'agit d'un statut de terminal.
Annulée	La commande a été annulée avant de se terminer. Il s'agit d'un état final.
Undeliverable (Non livrable)	La commande ne peut pas être délivrée au nœud géré. Le nœud peut ne pas exister ou ne peut pas répondre. Les appels ne pouvant pas être remis ne sont pas comptabilisés dans la limite `max-errors` de la commande parent, mais ils permettent de déterminer 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 5 appels, dont 4 renvoient le statut `Undeliverable` et 1 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 statut de terminal.

Statut	Détails
InvalidPlatform	La commande a été envoyée à un nœud géré qui ne correspondait pas aux plateformes requises spécifiées par le document choisi. `Invalid Platform` ne compte pas dans la limite maximale d'erreurs de la commande parent, mais permet de savoir si le statut de la commande parent est Success (Réussite) ou Failed (Échec). 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 5 appels, dont 4 renvoient le statut `Invalid Platform` et 1 renvoie le statut `Success`, le statut de la commande parent est `Success`. Il s'agit d'un statut de terminal.
AccessDenied	L'utilisateur ou le rôle AWS Identity and Access Management (IAM) à l'origine de la commande n'a pas accès au nœud géré ciblé. `Access Denied`n'est pas prise en compte dans la `max-errors` limite de la commande parent, mais elle contribue à détermine r si le statut de la commande parent est `Success` ou`Failed`. Par exemple, si tous les appels d'une commande ont le statut `Access Denied`, le statut de la commande renvoyé est `Failed`. Toutefois, si une commande comporte 5 appels, dont 4 renvoient le statut `Access Denied` et 1 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	Détails
En attente	La commande n'a pas encore été reçue par un agent sur un nœud géré.
InProgress	La commande a été envoyée au moins à un nœud géré, mais n'a pas atteint un état final sur tous les nœuds.
Delayed (Retardé)	Le système tente d'envoyer la commande au nœud, mais n'a pas réussi. Le système réessaie.
Réussite	La commande a été reçue par SSM Agent sur l'ensemble des nœuds gérés spécifiés ou ciblés 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 nœuds gérés spécifiés ou ciblés. 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 exception s en retournant les codes de sortie appropriés (codes de sortie autre que zéro pour un échec de la commande).
DeliveryTimedDehors	La commande n'a pas été fournie au nœud géré avant l'expiration du délai total. La valeur

Statut	Détails
	de `max-errors` ou d'autres appels de commande affichent le statut `Delivery Timed Out`. Il s'agit d'un état final.
Échec	La commande n'a pas réussi sur le nœud géré. La valeur de `max-errors` ou d'autres appels de commande affichent le statut `Failed`. Il s'agit d'un état final.
Incomplete (Incomplet)	La commande a été tentée sur tous les nœuds gérés et un ou plusieurs appels n'ont pas la valeur `Success`. Toutefois, le nombre d'appels en échec n'est pas suffisant pour que le statut soit `Failed`. Il s'agit d'un statut de terminal.
Annulée	La commande a été annulée avant de se terminer. Il s'agit d'un statut de terminal.
RateExceeded	Le nombre de nœuds gérés ciblée par la commande a dépassé la quota du compte pour les appels en attente. Le système a annulé la commande avant de l'exécuter sur un nœud. Il s'agit d'un statut de terminal.

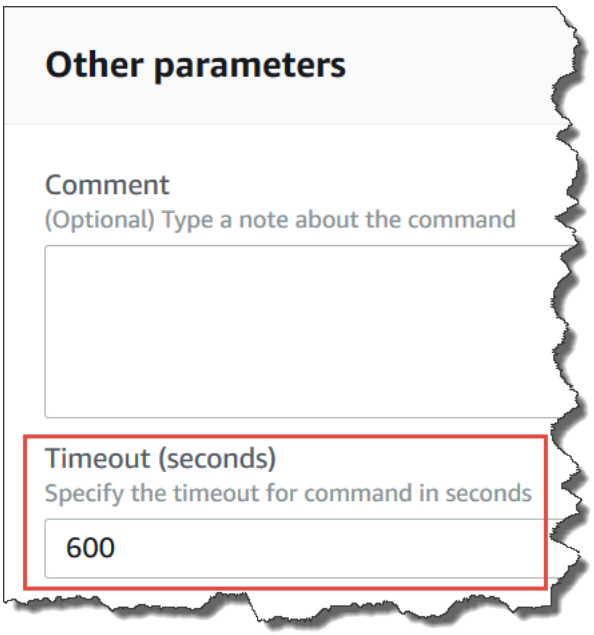
Statut	Détails
AccessDenied	L'utilisateur ou le rôle qui initie la commande n'a pas accès au groupe de ressources ciblé. `AccessDenied` n'est pas pris en compte dans la limite `max-errors` de la commande parent, mais contribue à déterminer si le statut de la commande parent est `Success` ou `Failed`. (Par exemple, si tous les appels d'une commande ont le statut `AccessDenied`, alors le statut de la commande retourné 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 aucun nœud géré. 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.

Total Timeout (Délai total)

Dans la console Systems Manager, vous spécifiez la valeur du délai d'expiration dans le champ Timeout (seconds) (Délai d'expiration (secondes)). Une fois qu'une commande est envoyée, Run Command vérifie si la commande a expiré ou non. Si une commande atteint la limite d'expiration de la commande (délai total), son statut devient `DeliveryTimedOut` pour tous les appels ayant le statut `InProgress`, `Pending` ou `Delayed`.



Sur un plan plus technique, le délai d'expiration total (Timeout (seconds) (Délai d'expiration (secondes))) est une combinaison de deux valeurs de délai d'expiration, comme indiqué ici :

`Total timeout = "Timeout(seconds)" from the console + "timeoutSeconds": "{{ executionTimeout }}" from your SSM document`

Par exemple, la valeur par défaut de Timeout (seconds) (Délai d'expiration (secondes)) dans la console Systems Manager est de 600 secondes. Si vous exécutez une commande en utilisant le document SSM `AWS-RunShellScript`, la valeur par défaut de « timeoutSeconds » : « {{executionTimeout}} » est de 3600 secondes, comme indiqué dans l'exemple de document suivant :

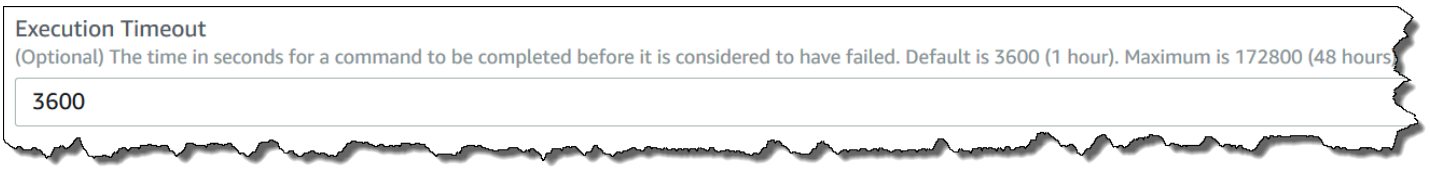
```
"executionTimeout": {
    "type": "String",
    "default": "3600",

"runtimeConfig": {
  "aws:runShellScript": {
    "properties": [
      {
        "timeoutSeconds": "{{ executionTimeout }}"
```

Cela signifie que la commande s'exécute pendant 4 200 secondes (70 minutes) avant que le système ne définisse l'état de la commande sur `DeliveryTimedOut`.

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. Le champ Execution Timeout (Délai d'exécution) n'est affiché que lorsqu'un paramètre d'entrée correspondant a été défini dans le document SSM. Si un délai est spécifié, la commande doit être exécutée dans ce délai.

> **Note**
>
> Run Command s'appuie sur la réponse terminale du document SSM Agent pour déterminer si la commande a été remise ou non à l'agent. SSM Agent doit envoyer un signal `ExecutionTimed0ut` pour qu'une invocation ou une commande soient marquées comme `ExecutionTimed0ut`.



Default Execution Timeout (Délai d'exécution par défaut)

Si un document 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.

Signalement des délais d'expiration par Systems Manager

Si Systems Manager reçoit une réponse `execution timeout` de l'SSM Agent sur une cible, Systems Manager marque l'invocation de commande comme `executionTimeout`.

Si Run Command ne reçoit pas de réponse terminale de document en provenance de l'SSM Agent, l'invocation de la commande est marquée comme `deliveryTimeout`.

Afin de déterminer le statut du délai sur une cible, l'SSM Agent combine tous les paramètres et le contenu du document SSM pour calculer `executionTimeout`. Lorsque l'SSM Agent détermine que le délai d'exécution d'une commande a expiré, il envoie `executionTimeout` au service.

La valeur par défaut pour Timeout (seconds) (Délai d'expiration (secondes)) est de 3 600 secondes. La valeur par défaut pour Execution timeout (Délai d'exécution) est également de 3 600 secondes. Par conséquent, le délai d'attente total par défaut pour une commande est de 7 200 secondes.

> **Note**
>
> L'SSM Agent traite `executionTimeout` différemment selon le type de document SSM et la version du document.

Procédures Run Command

Les procédures de cette section vous montrent comment exécuter des commandes avec Run Command, une des fonctionnalités de AWS Systems Manager, à l'aide de l'AWS Command Line Interface (AWS CLI) ou des AWS Tools for Windows PowerShell.

Table des matières

Vous pouvez également consulter des exemples de commandes dans les références suivantes.

- Référence de AWS CLI Systems Manager
- AWS Tools for Windows PowerShell - AWS Systems Manager

Mise à jour du logiciel à l'aide de Run Command

Les procédures suivantes décrivent comment mettre à jour le logiciel sur vos nœuds gérés.

Mise à jour de SSM Agent à l'aide de Run Command

La procédure suivante décrit comment mettre à jour l'SSM Agent en cours d'exécution sur vos nœuds gérés. Vous pouvez mettre à jour à l'aide de la dernière version de l'SSM Agent ou d'une version plus ancienne. Lorsque vous exécutez la commande, le système télécharge la version depuis 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**
>
> Si une instance exécute macOS version 11.0 (Big Sur) ou ultérieure, elle doit disposer de l'SSM Agent version 3.1.941.0 ou supérieure pour exécuter le document AWS-UpdateSSMAgent. Si l'instance exécute une version de l'SSM Agent publiée avant la version 3.1.941.0, vous pouvez mettre à jour votre SSM Agent pour exécuter le document AWS-UpdateSSMAgent en exécutant les commandes `brew update` et `brew upgrade amazon-ssm-agent`.

Pour être informé des SSM Agent mises à jour, abonnez-vous à la page des notes de SSM Agent publication surGitHub.

Pour mettre à jour l'SSM Agent en utilisant Run Command

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Run Command.
3. Sélectionnez Run Command (Exécuter la commande).
4. In the Command document (Document de commande), sélectionnez **`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, saisissez la version de l'SSM Agent 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), sélectionnez true pour installer une version antérieure de l'SSM Agent. Si vous sélectionnez cette option, spécifiez le numéro de version antérieure. Sélectionnez false pour installer uniquement la dernière version du service.
6. Dans la section Targets (Cibles), sélectionnez les nœuds gérés sur lesquels vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant des instances ou des appareils de périphérie manuellement ou en spécifiant un groupe de ressources.

> **Tip**
>
> Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés 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. Pour Rate control (Contrôle de débit) :

   - Dans Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage de nœuds gérés sur lesquels exécuter simultanément la commande.

     > **Note**
     >
     > Si vous avez sélectionné des cibles en spécifiant les balises appliquées aux nœuds gérés ou en spécifiant des groupes de ressources AWS , et que vous n'êtes pas certain du nombre de nœuds gérés ciblés, limitez le nombre de cibles autorisées à exécuter simultanément le document en indiquant un pourcentage.

   - Dans Error threshold (Seuil d'erreur), indiquez quand arrêter l'exécution de la commande sur les autres nœuds gérés après l'échec de celle-ci sur un certain nombre ou un certain pourcentage de nœuds. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les nœuds gérés sur lesquels la commande est toujours 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). Saisissez 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 (pour les instances EC2) ou de la

fonction du service IAM (pour les machines activées par un système hybride) attribués à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, assurez-vous que le profil d'instance ou le rôle de service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

10. Dans la section SNS notifications (Notifications SNS), si vous souhaitez envoyer des notifications sur le statut d'exécution des commandes, cochez la case Enable SNS notifications (Activer les notifications SNS).

    Pour plus d'informations sur la configuration des notifications Amazon SNS pour Run Command, consultez Surveillance des changements d'état du Systems Manager à l'aide des notifications Amazon SNS.

11. Cliquez sur Exécuter.

Mise à jour PowerShell en utilisant Run Command

La procédure suivante décrit comment effectuer une mise à jour PowerShell vers la version 5.1 sur vos nœuds gérés Windows Server 2012 et 2012 R2. Le script fourni dans cette procédure télécharge la mise à jour de Windows Management Framework (WMF) version 5.1 et démarre l'installation de la mise à jour. Le nœud redémarre au cours de ce processus, comme l'exige l'installation de WMF 5.1. Le téléchargement et l'installation de la mise à jour prennent environ cinq minutes.

Pour effectuer une mise à jour PowerShell avec Run Command

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Run Command.
3. Sélectionnez Run Command (Exécuter la commande).
4. In the Command document (Document de commande), sélectionnez **`AWS-RunPowerShellScript`**.
5. Dans la section Commands (Commandes), 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 Targets (Cibles), sélectionnez les nœuds gérés sur lesquels vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant des instances ou des appareils de périphérie manuellement ou en spécifiant un groupe de ressources.

> **Tip**
>
> Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés 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. Pour Rate control (Contrôle de débit) :

   - Dans Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage de nœuds gérés sur lesquels exécuter simultanément la commande.

> **Note**
>
> Si vous avez sélectionné des cibles en spécifiant les balises appliquées aux nœuds gérés ou en spécifiant des groupes de ressources AWS , et que vous n'êtes pas certain du nombre de nœuds gérés ciblés, limitez le nombre de cibles autorisées à exécuter simultanément le document en indiquant un pourcentage.

- Dans Error threshold (Seuil d'erreur), indiquez quand arrêter l'exécution de la commande sur les autres nœuds gérés après l'échec de celle-ci sur un certain nombre ou un certain pourcentage de nœuds. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les nœuds gérés sur lesquels la commande est toujours 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). Saisissez 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 (pour les instances EC2) ou de la fonction du service IAM (pour les machines activées par un système hybride) attribués à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, assurez-vous que le profil d'instance ou le rôle de service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

10. Dans la section SNS notifications (Notifications SNS), si vous souhaitez envoyer des notifications sur le statut d'exécution des commandes, cochez la case Enable SNS notifications (Activer les notifications SNS).

    Pour plus d'informations sur la configuration des notifications Amazon SNS pour Run Command, consultez Surveillance des changements d'état du Systems Manager à l'aide des notifications Amazon SNS.

11. Cliquez sur Exécuter.

Une fois que le nœud géré a redémarré et que l'installation de la mise à jour est terminée, connectez-vous à votre nœud pour confirmer que la mise à niveau vers la version 5.1 a été effectuée PowerShell avec succès. Pour vérifier la version de PowerShell sur votre nœud, ouvrez PowerShell et entrez`$PSVersionTable`. La valeur `PSVersion` dans le tableau de sortie affiche 5.1 si la mise à niveau a réussi.

Si la valeur `PSVersion` est différente de 5.1, par exemple 3.0 ou 4.0, consultez les journaux Setup (Configuration) dans Event Viewer, sous Windows Logs (Journaux Windows). Ces journaux indiquent la raison de l'échec de la mise à jour.

Procédure : utiliser les AWS CLI avec Run Command

L'exemple de procédure suivant vous explique comment utiliser l'AWS Command Line Interface (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 AWS Systems Manager illustrés dans cette rubrique. Les commandes ou scripts spécifiés dans des documents Systems Manager sont exécutés avec des autorisations administratives sur vos nœuds gérés. 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 au nœud. Pour tous les autres utilisateurs, vous devez créer des documents restrictifs et les partager avec des utilisateurs spécifiques.

Rubriques

Étape 1 : Démarrage

Vous devez disposer de autorisations administrateur sur les nœuds gérés que vous souhaitez configurer ou vous devez bénéficier de l'autorisation appropriée dans AWS Identity and Access Management (IAM). Notez également que cet exemple utilise la région USA Est (Ohio) (us-east-2). Run Command est disponible dans les Régions AWS répertoriées dans Points de terminaison de service Systems Manager dans la Référence générale d'Amazon Web Services. Pour de plus amples informations, veuillez consulter Con AWS Systems Manager figuration.

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 Command Line Interface (AWS CLI).

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l'AWS CLI.
2. Répertoriez 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'un nœud géré est prêt à recevoir les commandes.

   La sortie de la commande suivante indique si les nœuds gérés sont en ligne.

   Linux & macOS

   ```
   aws ssm describe-instance-information \
       --output text --query "InstanceInformationList[*]"
   ```

   Windows

   ```
   aws ssm describe-instance-information ^
       --output text --query "InstanceInformationList[*]"
   ```
4. Exécutez la commande suivante pour afficher les détails sur un nœud géré spécifique.

> **Note**
>
> Pour exécuter les commandes de cette procédure, remplacez l'instance et les ID de commande. Pour les appareils Core AWS IoT Greengrass gérés, utilisez mi-*ID_number* comme ID d'instance. L'ID de commande est renvoyé en réponse à send-command. Les ID d'instance sont disponibles depuis Fleet Manager, une fonctionnalité de AWS Systems Manager.

Linux & macOS

```
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

Run Command et le document `AWS-RunShellScript` vous permettent d'exécuter n'importe quel script ou commande sur un nœud géré comme si vous vous étiez connecté localement.

Afficher la description et les paramètres disponibles

Exécutez la commande suivante pour afficher la description du document JSON Systems Manager.

Linux & macOS

```
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 & macOS

```
aws ssm describe-document \
    --name "AWS-RunShellScript" \
    --query "Document.Parameters[*]"
```

Windows

```
aws ssm describe-document ^
    --name "AWS-RunShellScript" ^
    --query "Document.Parameters[*]"
```

Étape 3 : Envoyer des commandes simples à l'aide du document **`AWS-RunShellScript`**

Utilisez la commande suivante afin d'obtenir des informations IP pour un nœud géré Linux.

Si vous ciblez un nœud géré Windows Server, remplacez `document-name` par `AWS-RunPowerShellScript` et `command` depuis `ifconfig` par `ipconfig`.

Linux & macOS

```
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 & macOS

```
aws ssm list-command-invocations \
    --command-id $sh-command-id \
    --details
```

Windows

```
aws ssm list-command-invocations ^
    --command-id $sh-command-id ^
    --details
```

Identification de l'utilisateur

La commande suivante affiche l'utilisateur par défaut qui exécute les commandes.

Linux & macOS

```
sh_command_id=$(aws ssm send-command \
    --instance-ids "instance-ID" \
    --document-name "AWS-RunShellScript" \
    --comment "Demo run shell script on Linux managed node" \
    --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 le nœud géré. Cet exemple utilise l'ID de commande qui a été renvoyé lors de la commande précédente.

Linux & macOS

```
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 nœud géré.

Linux & macOS

```
aws ssm list-command-invocations \
    --command-id "command-ID" \
    --details
```

Windows

```
aws ssm list-command-invocations ^
    --command-id "command-ID" ^
    --details
```

Obtention d'informations sur la commande avec des données de réponse pour un nœud géré

La commande suivante renvoie la sortie de la demande `aws ssm send-command` initiale pour un nœud géré spécifique.

Linux & macOS

```
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 un nœud.

Linux & macOS

```
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 en utilisant Run Command

La commande suivante exécute un simple script Python « Hello World » à l'aide de Run Command.

Linux & macOS

```
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 : exécutez un script Bash avec Run Command

Les exemples de cette section montrent l'exécution du script bash suivant avec Run Command.

Pour obtenir des exemples d'utilisation de Run Command pour exécuter des scripts stockés dans des emplacements distants, consultez Exécution de scripts à partir d'Amazon S3 et Exécution de scripts depuis GitHub.

```
#!/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 l'agent AWS CodeDeploy sur les instances Amazon Linux et Red Hat Enterprise Linux (RHEL), comme décrit dans Créer une instance Amazon EC2 pour CodeDeploy dans le Guide de l'utilisateur AWS CodeDeploy.

Le script installe l'agent CodeDeploy à partir d'un compartiment S3 géré par AWS dans la région USA Est (Ohio) (us-east-2), `aws-codedeploy-us-east-2`.

Exécuter un script bash dans une commande AWS CLI

L'exemple suivant montre comment inclure le script bash dans une commande CLI en utilisant l'option `--parameters`.

Linux & macOS

```
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 en utilisant l'option `--cli-input-json`.

Linux & macOS

```
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
```

L'exemple suivant montre le contenu du fichier `installCodeDeployAgent.json` 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 pas à pas : utilisez le AWS Tools for Windows PowerShell avec Run Command

Les exemples suivants montrent comment utiliser le AWS Tools for Windows PowerShell pour afficher des informations sur les commandes et leurs paramètres, comment exécuter des commandes et comment afficher le statut de ces commandes. Cette procédure inclut un exemple pour chaque document AWS 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 une autorisation administrative sur vos nœuds gérés. Si un utilisateur est autorisé à exécuter l'un des documents prédéfinis de Systems Manager (tout document commençant par AWS), il dispose également d'un accès administrateur au nœud. Pour tous les autres utilisateurs, vous devez créer des documents restrictifs et les partager avec des utilisateurs spécifiques.

Rubriques

Configuration des paramètres AWS Tools for Windows PowerShell de session

Spécifier vos informations d'identification

Ouvrez Outils pour Windows PowerShell sur votre ordinateur local et exécutez la commande suivante pour spécifier vos informations d'identification. Vous devez disposer des autorisations d'administrateur sur les nœuds gérés que vous souhaitez configurer ou vous devez avoir obtenu les autorisations appropriées dans AWS Identity and Access Management (IAM). Pour plus d'informations, consultez Con AWS Systems Manager figuration.

```
Set-AWSCredentials -AccessKey key-name -SecretKey key-name
```

Définir une valeur par défaut Région AWS

Exécutez la commande suivante pour définir la région de votre PowerShell session. L'exemple utilise la région USA Est (Ohio) (us-east-2). Run Commandest disponible dans les points de terminaison du service Systems Manager Régions AWS répertoriés dans le Référence générale d'Amazon Web Services.

```
Set-DefaultAWSRegion `
    -Region us-east-2
```

Répertorier tous les documents disponibles

Cette commande répertorie les documents disponibles pour votre compte.

```
Get-SSMDocumentList
```

Exécuter PowerShell des commandes ou des scripts

Run Command et le document `AWS-RunPowerShell` vous permettent d'exécuter n'importe quel script ou commande sur un nœud géré comme si vous vous étiez connecté localement. Vous pouvez émettre des commandes ou saisir un chemin d'accès à un script local pour exécuter la commande.

> **Note**
>
> Pour plus d'informations sur le redémarrage des nœuds gérés lors de l'utilisation de Run Command pour appeler des scripts, consultez Gestion des redémarrages lors de l'exécution de commandes.

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 en utilisant le document **AWS-RunPowerShellScript**

La commande suivante permet d'afficher le contenu du répertoire "C:\Users" et celui du répertoire "C:\" sur deux nœuds gérés.

```
$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 nœuds gérés. Cet exemple utilise l'CommandId qui a été renvoyé lors de la commande précédente.

```
Get-SSMCommand `
    -CommandId $runPSCommand.CommandId
```

Dans cet exemple, le statut de la commande peut être Success, Pending ou InProgress.

Obtention d'informations sur la commande par nœud géré

La commande suivante utilise l'CommandId de la commande précédente afin d'obtenir le statut de l'exécution de la commande par nœud géré.

```
Get-SSMCommandInvocation `
    -CommandId $runPSCommand.CommandId
```

Obtention d'informations sur la commande avec des données de réponse pour un nœud géré

La commande suivante renvoie la sortie de la commande Send-SSMCommand initiale pour un nœud géré 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 en utilisant le document **`AWS-InstallApplication`**

En utilisant Run Command et le document `AWS-InstallApplication`, vous pouvez installer, réparer ou désinstaller des applications sur des nœuds gérés. Cette commande a besoin d'un chemin d'accès ou d'une adresse pour un MSI.

> **Note**
>
> Pour plus d'informations sur le redémarrage des nœuds gérés lors de l'utilisation de Run Command pour appeler des scripts, consultez Gestion des redémarrages lors de l'exécution de commandes.

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 en utilisant le document **AWS-InstallApplication**

La commande suivante installe une version de Python sur votre nœud géré 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'}
```

Obtention d'informations sur la commande par nœud géré

La commande suivante utilise l'`CommandId` afin d'obtenir le statut de l'exécution de la commande

```
Get-SSMCommandInvocation `
    -CommandId $installAppCommand.CommandId `
    -Details $true
```

Obtention d'informations sur la commande avec des données de réponse pour un nœud géré

La commande suivante retourne les résultats de l'installation Python.

```
Get-SSMCommandInvocation `
    -CommandId $installAppCommand.CommandId `
    -Details $true `
    -InstanceId instance-ID | Select -ExpandProperty CommandPlugins
```

Installation d'un PowerShell module à l'aide du document **AWS-InstallPowerShellModule** JSON

Vous pouvez l'utiliser Run Command pour installer PowerShell des modules sur des nœuds gérés. Pour plus d'informations sur les PowerShell modules, consultez la section PowerShell Modules 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
```

Installation d'un PowerShell module

La commande suivante permet de télécharger le fichier EZOut.zip, de l'installer, puis d'exécuter une commande supplémentaire afin d'installer XPS Viewer. Enfin, la sortie de cette commande est téléchargée 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
```

Obtention d'informations sur la commande par nœud géré

La commande suivante utilise l'`CommandId` afin d'obtenir le statut de l'exécution de la commande

```
Get-SSMCommandInvocation `
    -CommandId $installPSCommand.CommandId `
    -Details $true
```

Obtention d'informations sur la commande avec des données de réponse pour le nœud géré

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
```

Association d'un nœud géré à un domaine en utilisant le document JSON **AWS-JoinDirectoryServiceDomain**

En utilisantRun Command, vous pouvez rapidement joindre un nœud géré à un AWS Directory Service domaine. Avant d'exécuter cette commande, créez un répertoire. Nous vous recommandons également d'en découvrir plus sur l' AWS Directory Service. Pour plus d'informations, consultez le Guide d'administration AWS Directory Service.

Vous pouvez uniquement associer un nœud géré à un domaine. Vous ne pouvez pas supprimer de nœud d'un domaine.

> **Note**
>
> Pour plus d'informations sur les nœuds gérés lors de l'utilisation de Run Command pour appeler des scripts, consultez Gestion des redémarrages lors de l'exécution de commandes.

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
```

Association d'un nœud géré à un domaine

La commande suivante joint un nœud géré au AWS Directory Service domaine donné et télécharge toute sortie générée dans l'exemple de bucket Amazon Simple Storage Service (Amazon S3).

```
$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
```

Obtention d'informations sur la commande par nœud géré

La commande suivante utilise l'`CommandId` afin d'obtenir le statut de l'exécution de la commande

```
Get-SSMCommandInvocation `
    -CommandId $domainJoinCommand.CommandId `
    -Details $true
```

Obtention d'informations sur la commande avec des données de réponse pour le nœud géré

Cette commande renvoie la sortie du `Send-SSMCommand` initial pour le `CommandId` spécifique.

```
Get-SSMCommandInvocation `
    -CommandId $domainJoinCommand.CommandId `
    -Details $true | Select -ExpandProperty CommandPlugins
```

Envoyez les métriques Windows à Amazon CloudWatch Logs à l'aide du **`AWS-ConfigureCloudWatch`** document

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) à 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 les données des compteurs de performance à CloudWatch Logs.

Si vous avez précédemment activé CloudWatch l'intégration dans EC2Config, les paramètres de Systems Manager remplacent tous les paramètres stockés localement sur le nœud géré 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 de performance et les journaux sur un seul nœud géré, consultez la section Collecte de métriques et de journaux à partir d'instances Amazon EC2 et de serveurs sur site avec l'agent CloudWatch dans le guide de l'utilisateur Amazon. CloudWatch

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
```

Envoyez les journaux des applications à CloudWatch

La commande suivante configure le nœud géré et déplace les journaux des applications Windows vers 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.CloudWa
 "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"]}}}'}
```

Obtention d'informations sur la commande par nœud géré

La commande suivante utilise l'`CommandId` afin d'obtenir le statut de l'exécution de la commande

```
Get-SSMCommandInvocation `
    -CommandId $cloudWatchCommand.CommandId `
    -Details $true
```

Obtention d'informations sur la commande avec des données de réponse pour un nœud géré

La commande suivante renvoie les résultats de la CloudWatch configuration Amazon.

```
Get-SSMCommandInvocation `
    -CommandId $cloudWatchCommand.CommandId `
    -Details $true `
    -InstanceId instance-ID | Select -ExpandProperty CommandPlugins
```

Envoyer des compteurs de performance à CloudWatch l'utilisation du document **AWS-ConfigureCloudWatch**

La commande de démonstration suivante télécharge les compteurs de performance vers. CloudWatch Pour plus d'informations, consultez le guide de CloudWatch l'utilisateur Amazon.

```
$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.PerformanceCounterInputComp
 "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.Cl
 "Parameters":{"AccessKey":"", "SecretKey":"","Region":"region", "NameSpace":"Windows-
Default"}}], "Flows":{"Flows":["PerformanceCounter,CloudWatch"]}}}'}
```

Mettre à jour EC2Config en utilisant le document **AWS-UpdateEC2Config**

La fonctionnalité Run Command et le document `AWS-EC2ConfigUpdate` vous permettent de mettre à jour le service EC2Config qui s'exécute sur vos nœuds gérés Windows Server. Cette commande peut mettre à jour le service EC2Config à l'aide de la dernière version ou de la 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
```

Mettre à jour EC2Config à l'aide de la dernière version

```
$ec2ConfigCommand = Send-SSMCommand `
    -InstanceId instance-ID `
    -DocumentName "AWS-UpdateEC2Config"
```

Obtention d'informations sur la commande avec des données de réponse pour le nœud géré

Cette commande renvoie la sortie de la commande spécifiée depuis la commande `Send-SSMCommand` précédente.

```
Get-SSMCommandInvocation `
    -CommandId $ec2ConfigCommand.CommandId `
    -Details $true `
    -InstanceId instance-ID | Select -ExpandProperty CommandPlugins
```

Mettre à jour EC2Config à l'aide d'une version donnée

La commande suivante remplace EC2Config par ne version plus ancienne sur l'instance.

```
Send-SSMCommand `
    -InstanceId instance-ID `
    -DocumentName "AWS-UpdateEC2Config" `
    -Parameter @{'version'='4.9.3519'; 'allowDowngrade'='true'}
```

Activer ou désactiver la mise à jour automatique de Windows en utilisant le document **`AWS-ConfigureWindowsUpdate`**

Run Command et le document `AWS-ConfigureWindowsUpdate` vous permettent d'activer et de désactiver les mises à jour Windows automatiques sur vos nœuds gérés 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, le nœud géré 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 de 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 22 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 de 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 le nœud géré 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 activer 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 Run Command

Run Command et le document `AWS-InstallWindowsUpdates` vous permettent de gérer les mises à jour pour les nœuds gérés Windows Server. Cette commande recherche ou installe les mises à jour manquantes sur vos nœuds gérés et, éventuellement, provoque un redémarrage après l'installation. Vous pouvez également spécifier les classifications et les niveaux de sévérité appropriés pour les mises à jour à installer dans votre environnement.

> **Note**
>
> Pour plus d'informations sur le redémarrage des nœuds gérés lors de l'utilisation de Run Command pour appeler des scripts, consultez Gestion des redémarrages lors de l'exécution de commandes.

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'}
```

Résolution des problèmes liés à Run Command de Systems Manager

Run Command, une des fonctionnalités de AWS Systems Manager, 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. Vous pouvez également utiliser les informations de cette rubrique pour aider à la résolution les problèmes rencontrés avec la fonctionnalité Run Command.

Rubriques

- Certains de mes nœuds gérés sont manquants
- Une étape de mon script a échoué, mais l'état global est « réussi »
- SSM Agent ne fonctionne pas correctement

Certains de mes nœuds gérés sont manquants

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 nœuds gérés sur lesquels vous pouvez choisir d'exécuter la commande s'affiche.

Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés pour obtenir des conseils de dépannage.

Après avoir créé, activé, redémarré ou relancé un nœud géré, installé Run Command sur une nœud ou attaché un profil d'instance AWS Identity and Access Management (IAM) à un nœud, l'ajout du nœud géré à la liste peut prendre quelques minutes.

Une étape de mon script a échoué, mais l'état global est « réussi »

Vous pouvez utiliser Run Command pour définir la manière dont les codes de sortie sont gérés par 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 plus d'informations et d'exemples, consultez Spécifier les codes de sortie dans les commandes.

SSM Agent ne fonctionne pas correctement

Si vous rencontrez des problèmes pour exécuter des commandes avec la fonctionnalité Run Command, cela peut venir de l'SSM Agent. Pour obtenir des informations sur la recherche de problèmes avec SSM Agent, veuillez consulter Résolution des problèmes de SSM Agent.

AWS Systems Manager State Manager

State Manager, une fonctionnalité de AWS Systems Manager, est un service de gestion de configuration sécurisé et évolutif qui automatise le processus de maintien de vos nœuds gérés et autres AWS ressources dans un état que vous définissez. Pour vos premiers pas dans State Manager, ouvrez Systems Manager console. Dans le panneau de navigation, sélectionnez State Manager.

> **Note**
>
> State Manager et Maintenance Windows peuvent effectuer certains types de mises à jour similaires sur vos nœuds gérés. Votre choix dépend de la nécessité d'automatiser la conformité du système ou d'effectuer des tâches hautement prioritaires et sensibles au temps pendant les périodes que vous spécifiez.
> Pour plus d'informations, consultez Choisir entre State Manager et Maintenance Windows.

Comment mon organisation peut-elle tirer parti de State Manager ?

En utilisant des documents Systems Manager préconfigurés (documents SSM), State Manager offre les avantages suivants pour la gestion de vos nœuds :

- Amorcer les nœuds avec un logiciel spécifique au démarrage.
- Télécharger et mettre à jour les agents selon un programme défini, y compris l'SSM Agent.
- Configurer les paramètres réseau.
- Associer des nœuds à un domaine Microsoft Active Directory.
- Exécuter les scripts sur les nœuds gérés Linux, macOS et Windows tout au long de leur cycle de vie.

Pour gérer la dérive de configuration entre d'autres AWS ressources, vous pouvez utiliser Automation, une fonctionnalité de Systems Manager, State Manager pour effectuer les types de tâches suivants :

- Attachez un rôle Systems Manager à des instances Amazon Elastic Compute Cloud (Amazon EC2) pour en faire des nœuds gérés.
- Appliquer les règles d'entrée et de sortie souhaitées pour un groupe de sécurité.
- Créez ou supprimez des sauvegardes Amazon DynamoDB.
- Créez ou supprimez des instantanés Amazon Elastic Block Store (Amazon EBS).
- Désactivez les autorisations de lecture et d'écriture sur des compartiments Amazon Simple Storage Service (Amazon S3).
- Démarrez, redémarrez ou arrêtez des nœuds gérés et des nœuds Amazon Relational Database Service (Amazon RDS).
- Appliquez des correctifs à des AMIs Linux, macOS et Windows.

Pour plus d'informations sur l'utilisation de State Manager avec des runbooks Automation, consultez Exécution des automatisations avec les associations State Manager.

À qui est destiné State Manager ?

State Managerconvient à tous les AWS clients qui souhaitent améliorer la gestion et la gouvernance de leurs AWS ressources et réduire la dérive des configurations.

Quelles sont les fonctions d'State Manager ?

Les principales fonctionnalités de State Manager sont décrites ci-après.

- Associations dans State Manager

  Une State Manager association est une configuration que vous attribuez à vos AWS ressources. La configuration définit le statut que vous souhaitez conserver sur vos ressources. Par exemple, une association peut spécifier qu'un logiciel antivirus doit être installé et s'exécuter sur un nœud géré, ou que certains ports doivent être fermés.

  Une association spécifie une planification indiquant quand appliquer la configuration et quelles sont les cibles de l'association. Par exemple, une association pour un logiciel antivirus peut s'exécuter une fois par jour sur tous les nœuds gérés d'un Compte AWS. Si le logiciel n'est pas installé sur un

nœud, l'association pourrait demander à State Manager de l'installer. Si le logiciel est installé, mais que le service ne s'exécute pas, l'association pourrait demander à State Manager de démarrer le service.

- Options de planification flexibles

  State Manager offre les options suivantes de planification lorsqu'une association s'exécute :

  - Traitement immédiat ou retardé

    Lorsque vous créez une association, par défaut, le système l'exécute immédiatement sur les ressources spécifiées. Après l'exécution initiale, l'association s'exécute à des intervalles selon la planification que vous définissez.

    Vous pouvez demander à State Manager de ne pas exécuter immédiatement une association en utilisant l'option Apply association only at the next specified Cron interval (Appliquer l'association uniquement à l'intervalle Cron spécifié suivant) dans la console ou le paramètre `ApplyOnlyAtCronInterval` de la ligne de commande.

  - Expressions cron et rate

    Lorsque vous créez une association, vous spécifiez le moment où State Manager applique la configuration. State Manager prend en charge la plupart des expressions cron et rate standard pour la planification lorsqu'une association s'exécute. State Manager prend également en charge les expressions cron qui incluent un jour de la semaine et le signe numérique (#) pour désigner le nième jour d'un mois pour exécuter une association et le signe (L) indiquant le dernier X jour du mois.

    > **Note**
    >
    > State Manager ne prend actuellement pas en charge la spécification de mois dans les expressions cron pour les associations.

    Pour contrôler davantage l'exécution d'une association, par exemple si vous souhaitez exécuter une association deux jours après le correctif mardi, vous pouvez spécifier un décalage. Un offset (décalage) définit le nombre de jours d'attente après le jour prévu pour exécuter une association.

    Pour plus d'informations sur la génération d'expressions cron et rate, reportez-vous à Référence : Expressions Cron et Rate pour Systems Manager.

- Options de ciblage multiple

Une association spécifie également les cibles de l'association. State Managerprend en charge le ciblage AWS des ressources à l'aide de balises AWS Resource Groups, d'identifiants de nœuds individuels ou de tous les nœuds gérés dans le Région AWS et Compte AWS.

- Prise en charge d'Amazon S3

  Stockez les sorties de commandes des exécutions d'association dans le compartiment Amazon S3 de votre choix. Pour plus d'informations, consultez Utilisation d'associations dans Systems Manager.

- EventBridge soutien

  Cette fonctionnalité de Systems Manager est prise en charge à la fois en tant que type d'événement et en tant que type de cible dans EventBridge les règles Amazon. Pour plus d'informations, consultez Surveillance d'événements Systems Manager avec Amazon EventBridge et Référence : modèles et types d'événements Amazon EventBridge pour Systems Manager.

L'utilisation d'State Manager entraîne-t-elle des frais ?

State Manager est disponible sans frais supplémentaires.

Comment démarrer avec State Manager ?

Pour commencer à utiliser State Manager, procédez comme suit :

Tâche	Pour en savoir plus
Configuration de Systems Manager	Con AWS Systems Manager figuration
En savoir plus sur State Manager	A propos d'State Manager
Créez et attribuez une association State Manager à vos nœuds	Utilisation d'associations dans Systems Manager

Plus d'informations

- Lutter contre la dérive de configuration à l'aide d'Amazon EC2 Systems Manager et PowerShell de Windows DSC
- Configurer les instances Amazon EC2 dans un groupe Auto Scaling à l'aide de State Manager

Rubriques

A propos d'State Manager

State Manager, une fonctionnalité de AWS Systems Manager, est un service sécurisé et évolutif qui automatise le processus de maintien des nœuds gérés dans une infrastructure hybride et multicloud dans un état que vous définissez.

Voici comment State Manager fonctionne :

1. Déterminez l'état que vous souhaitez appliquer à vos AWS ressources.

   Voulez-vous garantir que vos nœuds gérés sont configurés avec des applications spécifiques, telles que des applications antivirus ou malveillantes ? Voulez-vous automatiser le processus de mise à jour de l'SSM Agent ou d'autres packages AWS comme `AWSPVDriver` ? Devez-vous vous assurer que des ports spécifiques sont fermés ou ouverts ? Pour commencerState Manager, déterminez l'état que vous souhaitez appliquer à vos AWS ressources. L'état que vous voulez appliquer détermine le document SSM que vous utilisez pour créer une association State Manager.

   Une State Manager association est une configuration que vous attribuez à vos AWS ressources. La configuration définit le statut que vous souhaitez conserver sur vos ressources. Par exemple, une association peut spécifier qu'un logiciel antivirus doit être installé et s'exécuter sur un nœud géré, ou que certains ports doivent être fermés.

   Une association spécifie une planification indiquant quand appliquer la configuration et quelles sont les cibles de l'association. Par exemple, une association pour un logiciel antivirus peut s'exécuter une fois par jour sur tous les nœuds gérés d'un Compte AWS. Si le logiciel n'est pas installé sur un nœud, l'association pourrait demander à State Manager de l'installer. Si le logiciel est installé, mais que le service ne s'exécute pas, l'association pourrait demander à State Manager de démarrer le service.

2. Déterminez si un document SSM préconfiguré peut vous aider à créer l'état souhaité sur vos AWS ressources.

   Systems Manager inclut des douzaines de documents SSM 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 d'Amazon CloudWatch, l'exécution d' AWS Systems Manager automatismes, l'exécution de scripts Shell PowerShell et l'association de nœuds gérés à un domaine de service d'annuaire pour Active Directory.

   Vous pouvez afficher tous les documents SSM dans la console Systems Manager. Sélectionnez le nom d'un document pour en savoir plus sur celui-ci. Voici deux exemples : `AWS-ConfigureAWSPackage` et `AWS-InstallApplication`.

3. Créer une association.

   Vous pouvez créer une association à l'aide de la console Systems Manager, de l'API AWS Command Line Interface AWS Tools for Windows PowerShell (AWS CLI), (Tools for Windows PowerShell) ou de l'API Systems Manager. Lorsque vous créez une association, vous spécifiez les informations suivantes :

   - Un nom pour l'application.
   - 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 nœuds).
   - Des cibles pour l'association. Vous pouvez cibler des nœuds gérés en spécifiant des identifications, en choisissant des ID de nœuds individuels ou en choisissant un groupe dans AWS Resource Groups. Vous pouvez également cibler tous les nœuds gérés dans les versions actuelles Région AWS et Compte AWS.
   - 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.

     > **Note**
     >
     > State Manager ne prend actuellement pas en charge la spécification de mois dans les expressions cron pour les associations.

   Lorsque vous exécutez la commande pour créer l'association, Systems Manager lie les informations que vous avez spécifiées (planification, cibles, documents SSM et paramètres)

aux ressources ciblées. 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 ressources. Vous pouvez consulter les détails du statut dans la console ou (pour les nœuds gérés) à l'aide de l'opération DescribeInstanceAssociationsStatusAPI. 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 compartiment Amazon Simple Storage Service (Amazon S3) spécifié.

Pour plus d'informations, consultez Utilisation d'associations dans Systems Manager.

> **Note**
>
> Les opérations d'API initiées par le document SSM lors d'une exécution d'association ne sont pas journalisées dans AWS CloudTrail.

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 Affichage des historiques des associations. 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 Modification et création de la nouvelle version d'une association.

Quand les associations sont-elles appliquées aux ressources ?

Lorsque vous créez une association, vous spécifiez un document SSM qui définit la configuration, une liste des ressources cibles et un calendrier pour l'application de la configuration. Par défaut, State Manager exécute l'association lorsque vous la créez, puis selon votre calendrier. State Manager tente également d'exécuter l'association dans les situations suivantes :

- Modification de l'association : State Manager exécute l'association après qu'un utilisateur a modifié et enregistré ses modifications dans l'un des champs d'association suivants : `DOCUMENT_VERSION`, `PARAMETERS`, `SCHEDULE_EXPRESSION`, `OUTPUT_S3_LOCATION`.
- Modification du document : State Manager exécute l'association après qu'un utilisateur a modifié et enregistré les modifications apportées au document SSM qui définit l'état de configuration de l'association. Plus précisément, l'association s'exécute une fois que les modifications suivantes ont été apportées au document :
  - Un utilisateur spécifie une nouvelle version de document $DEFAULT et l'association a été créée à l'aide de la version $DEFAULT.
  - Un utilisateur met à jour un document et l'association a été créée à l'aide de la version $LATEST.
  - Un utilisateur supprime le document qui a été spécifié lors de la création de l'association.
- Changement de valeur d'un paramètre de Parameter Store : State Manager exécute l'association après qu'un utilisateur a modifié la valeur d'un paramètre défini dans l'association.
- Démarrage manuel : State Manager exécute l'association lorsque l'utilisateur la lance à partir de la console Systems Manager ou par programmation.
- Modifications de cible : State Manager exécute l'association après que l'une des activités suivantes se soit produite sur un nœud cible :
  - Un nœud géré est mis en ligne pour la première fois.
  - Un nœud géré est mis en ligne après avoir manqué une exécution d'association planifiée.
  - Un nœud géré est mis en ligne après avoir été arrêté pendant plus de 30 jours.

> **Note**
>
> Les mises à jour de la cible n'affectent pas les associations créées à l'aide de Systems Manager Automation.

Utilisation d'associations dans Systems Manager

Cette section décrit comment créer et gérer des associations State Manager en utilisant la console AWS Systems Manager, l'AWS Command Line Interface (AWS CLI), et AWS Tools for PowerShell.

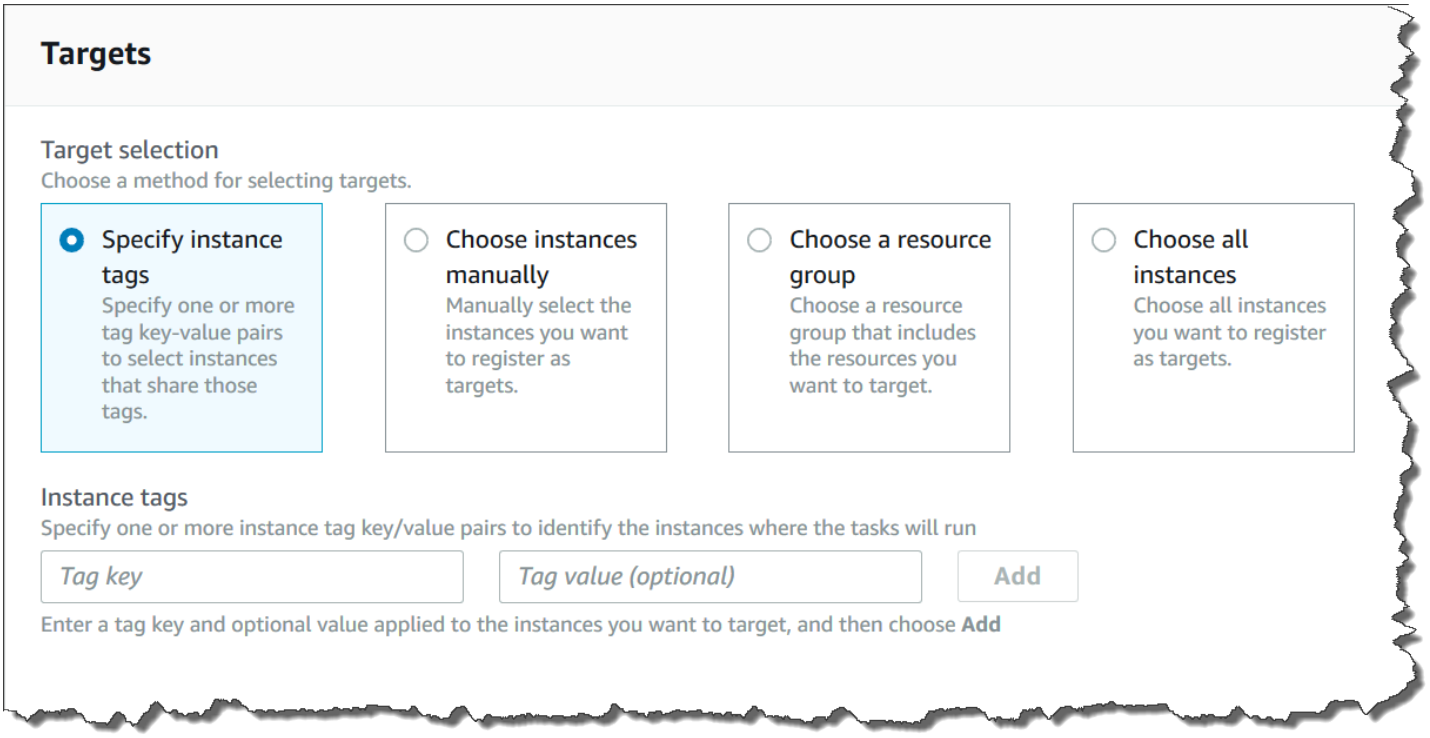
Rubriques

À propos des cibles et des contrôles du débit dans les associations State Manager

Cette rubrique décrit les fonctionnalités State Manager, une des fonctionnalités de AWS Systems Manager, qui vous aident à déployer une association sur des dizaines ou des centaines de nœuds, tout en contrôlant le nombre de nœuds qui exécutent l'association à l'heure prévue.

Cibles

Lorsque vous créez une association State Manager, vous sélectionnez les nœuds à configurer avec cette association 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 Command Line Interface (AWS CLI), vous spécifiez le paramètre `targets`. Le ciblage des nœuds vous permet de configurer des dizaines, des centaines ou des milliers de nœuds avec une association sans devoir spécifier ou choisir des ID de nœuds individuels.

Chaque nœud géré peut être ciblé par un maximum de 20 associations.

La fonctionnalité State Manager inclut les options cibles suivantes lors de la création d'une association.

Spécifier des identifications

Utilisez cette option pour spécifier une clé d'identification et (éventuellement) une valeur d'identification affectée à vos nœuds. Lorsque vous exécutez la demande, le système recherche et tente de créer l'association sur tous les nœuds qui correspondent à la clé et à la valeur d'identification spécifiées. Si vous avez spécifié plusieurs valeurs d'identification, l'association cible tous les nœuds contenant au moins une de ces valeurs d'identification. 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 des nœuds et affectez la clé d'identification et la valeur spécifiées à ces derniers, le système applique automatiquement l'association, l'exécute immédiatement, puis l'exécute

conformément à la planification. Cela s'applique lorsque l'association utilise un document Command ou Policy, mais pas lorsque l'association utilise un runbook Automation. Si vous supprimez les identifications spécifiées d'un nœud, le système n'exécute plus l'association sur ces nœuds.

> **Note**
>
> Si vous utilisez des runbooks Automation avec State Manager et que la limitation de l'identification vous empêche d'atteindre un objectif spécifique, pensez à utiliser des runbooks d'automatisation avec Amazon EventBridge. Pour de plus amples informations, veuillez consulter Exécution d'automatisations basées sur les événements. Pour plus d'informations sur l'utilisation de runbooks avec State Manager reportez-vous à Exécution des automatisations avec les associations State Manager.

Il est recommandé d'utiliser des balises lors de la création d'associations qui utilisent un document de commande ou de politique. Nous vous recommandons également d'utiliser des balises lors de la création d'associations pour exécuter des groupes Auto Scaling. Pour de plus amples informations, veuillez consulter Exécution de groupes Auto Scaling avec des associations.

> **Note**
>
> Notez les informations suivantes.
>
> - Lors de la création d'une association dans la console, lorsque vous ciblez des nœuds à l'aide de balises, vous ne pouvez spécifier qu'une seule clé de balise. Si vous souhaitez utiliser la console et cibler vos nœuds en utilisant plusieurs clés de balise, attribuez les clés de balise à un groupe AWS Resource Groups et ajoutez-y les nœuds. Vous pouvez ensuite choisir l'option Groupe de ressources dans la liste des cibles lorsque vous créez l'association State Manager.
> - Vous pouvez spécifier un maximum de cinq clés de balise en utilisant la AWS CLI. Si vous utilisez la AWS CLI, toutes les clés de balise spécifiées dans la commande `create-association` doivent être actuellement attribuées au nœud. Si elles ne le sont pas, State Manager ne parvient pas à cibler le nœud pour une association. Pour plus d'informations sur l'affectation d'identifications à vos nœuds, consultez Balisage des ressources Systems Manager.

Choix manuel des nœuds

Utilisez cette option pour sélectionner manuellement les nœuds dans lesquelles vous souhaitez créer l'association. Le panneau Instances affiche tous les nœuds gérés par Systems Manager dans l'Compte AWS et l'Région AWS actuels. Vous pouvez sélectionner manuellement autant de nœuds 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, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés pour obtenir des conseils de dépannage.

Pour choisir un groupe de ressources

Utilisez cette option pour créer une association sur tous les nœuds renvoyés par une requête basée sur une identification AWS Resource Groups ou une pile AWS CloudFormation.

Vous trouverez ci-dessous des informations détaillées sur le ciblage des groupes de ressources pour une association.

- Si vous ajoutez de nouveaux nœuds à un groupe, le système mappe automatiquement les nœuds à l'association qui cible le groupe de ressources. Le système applique l'association aux nœuds lorsqu'il détecte la modification. Après cette première exécution, le système exécute l'association selon la planification que vous avez spécifiée.
- Si vous créez une association qui cible un groupe de ressources et que le type de ressource `AWS::SSM::ManagedInstance` a été spécifié pour ce groupe, l'association s'exécute à la fois sur des instances Amazon Elastic Compute Cloud (Amazon EC2) et des nœuds non EC2 d'un environnement hybride et multicloud.
- Si vous créez une association qui cible un groupe de ressources, ce groupe ne doit pas disposer de plus de cinq clés de balises affectées à celui-ci ou plus de cinq valeurs spécifiées pour une clé de balises. Si l'une de ces conditions s'applique aux balises et aux clés attribuées à votre groupe de ressources, l'association ne s'exécutera pas et renverra une erreur `InvalidTarget`.
- 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 déterminer le statut de l'association sur 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.

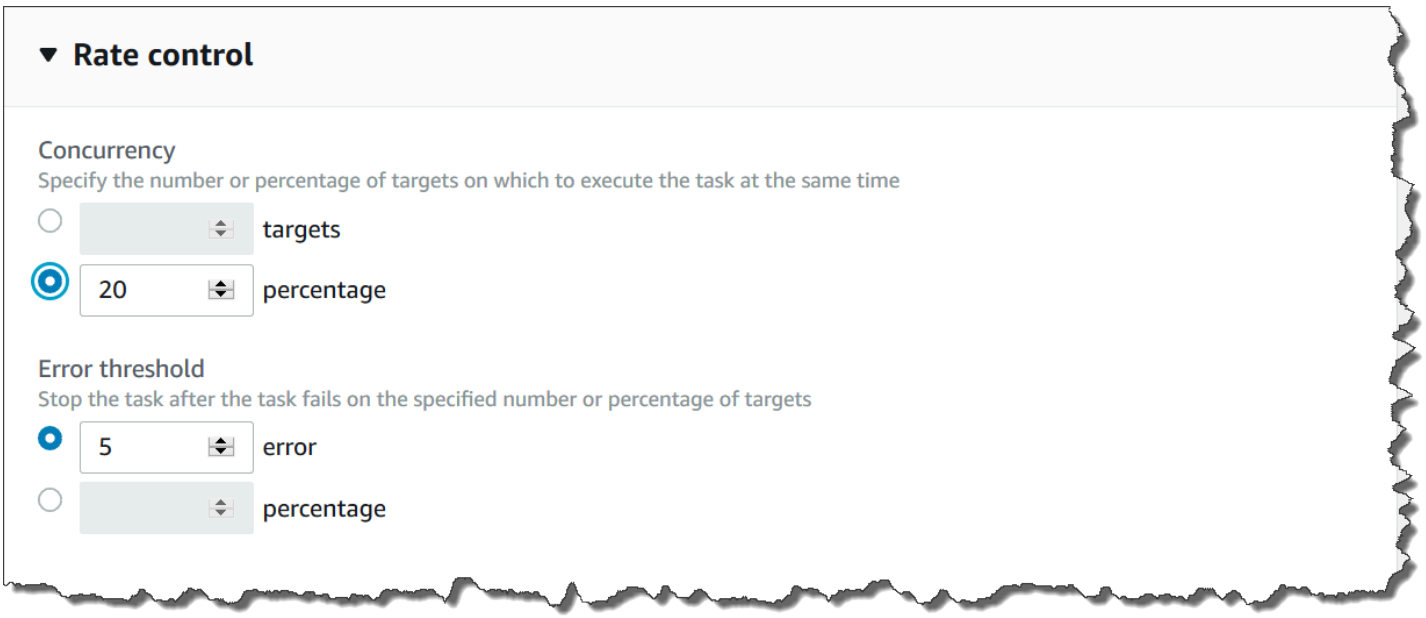
Pour de plus amples informations sur Resource Groups, consultez Qu'est-ce que AWS Resource Groups ? dans le Guide de l'utilisateur AWS Resource Groups.

Choisir tous les nœuds

Utilisez cette option pour cibler tous les nœuds de l'Compte AWS et de l'Région AWS actuels. Lorsque vous exécutez la demande, le système localise et tente de créer l'association sur tous les nœuds de l'Compte AWS et de l'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 des nœuds, 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 nœuds en spécifiant une valeur de simultanéité et un seuil d'erreur. La valeur de simultanéité spécifie le nombre de nœuds autorisés 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 nœud configuré avec cette association pour qu'il cesse 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.



Simultanéité

La simultanéité permet de limiter l'impact sur vos nœuds en vous permettant de spécifier que seul un certain nombre de nœuds peuvent traiter une association à la fois. Vous pouvez spécifier soit un nombre absolu de nœuds, par exemple 20, soit un pourcentage de l'ensemble cible de nœuds, 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 nœuds.
- Si de nouveaux nœuds correspondant aux critères cibles sont mis en ligne pendant qu'une association utilisant la simultanéité s'exécute, les nouveaux nœuds exécutent l'association si la valeur de simultanéité n'est pas dépassée. Si la valeur de la concurrence est dépassée, les nœuds sont ignorés pendant l'intervalle d'exécution de l'association en cours. Les nœuds 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'un ou plusieurs nœuds traitent cette association lorsque celle-ci est mise à jour, tout nœud qui exécute l'association sera autorisé à se terminer. Les associations qui n'ont pas commencé sont abandonnées. Une fois que les associations en cours d'exécution sont terminées, tous les nœuds 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 nœuds configuré 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.

Création d'associations

State Manager, une fonctionnalité de AWS Systems Manager, vous aide à maintenir vos AWS ressources dans un état qui vous permet de définir et de réduire la dérive de configuration. Pour ce faire, State Manager utilise des associations. Une association est une configuration que vous attribuez à vos AWS ressources. La configuration définit le statut que vous souhaitez conserver sur vos ressources. Par exemple, une association peut spécifier qu'un logiciel antivirus doit être installé et s'exécuter sur un nœud géré, ou que certains ports doivent être fermés.

Une association spécifie une planification indiquant quand appliquer la configuration et quelles sont les cibles de l'association. Par exemple, une association pour un logiciel antivirus peut s'exécuter une fois par jour sur tous les nœuds gérés d'un Compte AWS. Si le logiciel n'est pas installé sur un nœud, l'association pourrait demander à State Manager de l'installer. Si le logiciel est installé, mais que le service ne s'exécute pas, l'association pourrait demander à State Manager de démarrer le service.

> **Note**
>
> Vous pouvez attribuer des balises à une association lorsque vous la créez à l'aide d'un outil de ligne de commande tel que le AWS CLI ou AWS Tools for PowerShell. L'ajout de balises à une association à l'aide de la console Systems Manager n'est pas pris en charge. Pour en savoir plus sur les identifications, consultez Balisage des ressources Systems Manager.

Les procédures suivantes décrivent comment créer une association qui utilise une `Command` ou un document `Policy` pour cibler les nœuds gérés. Pour plus d'informations sur la création d'une association utilisant un runbook d'automatisation pour cibler des nœuds ou d'autres types de AWS ressources, consultezExécution des automatisations avec les associations State Manager.

Objectifs des associations et contrôles du débit

Une association spécifie également quels nœuds gérés, ou cibles, doivent recevoir l'association. State Manager inclut plusieurs fonctions pour vous aider à cibler vos nœuds gérés et à contrôler la façon dont l'association est déployée sur ces cibles. 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.

Exécuter les associations

Par défaut, State Manager exécute une association immédiatement après sa création, puis selon le calendrier que vous avez défini.

Le système exécute également les associations selon les règles suivantes :

- State Manager tente d'exécuter l'association sur tous les nœuds spécifiés ou ciblés 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 de nœuds pouvant traiter l'association simultanément), State Manager tente d'exécuter l'association lors du prochain intervalle.

- State Manager exécute l'association après avoir modifié la configuration, les nœuds cibles, les documents ou les paramètres de l'association. Pour de plus amples informations, consultez Quand les associations sont-elles appliquées aux ressources ?.
- State Manager enregistre un historique pour tous les intervalles ignorés. Vous pouvez consulter l'historique dans l'onglet Execution History (Historique d'exécution).

Planifier les associations

Vous pouvez planifier des associations pour qu'elles s'exécutent à des intervalles de base, par exemple toutes les 10 heures, ou vous pouvez créer des planifications plus avancées à l'aide d'expressions cron et rate personnalisées. Vous pouvez également empêcher les associations de s'exécuter lorsque vous les créez pour la première fois.

Utiliser les expressions cron et rate pour planifier les exécutions des associations

Outre les expressions cron ou rate standard, State Manager prend également en charge les expressions cron qui incluent un jour de la semaine et le signe numérique (#) pour désigner le ne jour d'un mois pour exécuter une association. Voici un exemple qui exécute une planification cron le troisième mardi de chaque mois à 23 h 30 UTC :

```
cron(30 23 ? * TUE#3 *)
```

Voici un exemple qui se déroule le deuxième jeudi de chaque mois à minuit UTC :

```
cron(0 0 ? * THU#2 *)
```

State Manager prend également en charge le signe (L) pour indiquer le dernier X jour du mois. Voici un exemple qui exécute une planification cron le dernier mardi de chaque mois à 23 h 30 UTC :

```
cron(0 0 ? * 3L *)
```

Pour contrôler davantage l'exécution d'une association, par exemple si vous souhaitez exécuter une association deux jours après le correctif mardi, vous pouvez spécifier un décalage. Un offset (décalage) définit le nombre de jours d'attente après le jour prévu pour exécuter une association. Par exemple, si vous avez spécifié une planification cron de `cron(0 0 ? * THU#2 *)`, vous pouvez spécifier le numéro 3 dans le champ Schedule offset (Décalage de planification) pour exécuter l'association tous les dimanches après le deuxième jeudi du mois.

> **Note**
>
> Pour utiliser des décalages, vous devez sélectionner Appliquer l'association uniquement à l'intervalle Cron spécifié suivant dans la console ou vous devez spécifier le paramètre `ApplyOnlyAtCronInterval` dans la ligne de commande. Lorsque l'une de ces options est activée, State Manager n'exécute pas d'association immédiatement après sa création.

Pour plus d'informations sur les expressions de type cron et rate, consultez Référence : Expressions Cron et Rate pour Systems Manager.

Créer une association (console)

La procédure suivante décrit comment utiliser la console Systems Manager pour créer une association State Manager.

> **Warning**
>
> Lorsque vous créez une association, vous pouvez choisir un groupe de AWS ressources de nœuds gérés comme cible pour l'association. Si un utilisateur, un groupe ou un rôle AWS Identity and Access Management (IAM) est autorisé à créer une association qui cible un groupe de ressources de nœuds gérés, cet utilisateur, ce groupe ou ce rôle contrôle automatiquement au niveau racine tous les nœuds du groupe. Seuls les administrateurs approuvés doivent être autorisés à créer des associations.

Pour créer une association State Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez State Manager.
3. Sélectionnez Créer une association.
4. Dans le champ Nom, spécifiez un nom.
5. Dans la liste Document, sélectionnez l'option en regard du nom d'un document. Notez le type de document. Cette procédure s'applique aux documents `Policy` et `Command`. Pour plus d'informations sur la création d'une association qui utilise un runbook Automation, consultez Exécution des automatisations avec les associations State Manager.

> **Important**
>
> State Manager ne prend pas en charge l'exécution d'associations utilisant une nouvelle version d'un document si ce document est partagé à partir d'un autre compte. State Manager exécute toujours la version `default` d'un document s'il est partagé depuis un autre compte, même si la console Systems Manager indique qu'une nouvelle version a été traitée. Si vous souhaitez exécuter une association à l'aide d'une nouvelle version d'un document partagé à partir d'un autre compte, vous devez définir la version du document sur `default`.

6. Pour Parameters (Paramètres), spécifiez les paramètres d'entrée requis.
7. (Facultatif) Choisissez une CloudWatch alarme à appliquer à votre association à des fins de surveillance.

> **Note**
>
> Notez les informations suivantes concernant ce passage.
>
> - La liste des alarmes affiche 100 alarmes maximum. Si votre alarme ne figure pas dans la liste, utilisez le AWS Command Line Interface pour créer l'association. Pour plus d'informations, consultez Créer une association (ligne de commande).
> - Pour associer une CloudWatch alarme à votre commande, le principal IAM qui crée l'association doit être autorisé à effectuer l'`iam:createServiceLinkedRole`action. Pour plus d'informations sur les CloudWatch alarmes, consultez la section Utilisation des CloudWatch alarmes Amazon.
> - Si votre alarme se déclenche, toute invocation ou automatisme de commande en attente ne s'exécute pas.

8. Pour Targets (Cibles), sélectionnez 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.
9. Dans la section Specify schedule (Spécifier le programme), sélectionnez On Schedule (Selon le calendrier) ou No schedule (Pas de calendrier). Si vous sélectionnez 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, sélectionnez Appliquer l'association uniquement à l'intervalle Cron spécifié suivant.

10. (Facultatif) Dans le Schedule offset (Décalage de planification), spécifiez un nombre compris entre 1 et 6.
11. Dans la section Options avancées utilisez Compliance severity (Sévérité de la conformité) pour choisir un niveau de sévérité pour l'association, et utilisez Change Calendriers (Calendriers de modifications) pour choisir un calendrier des modifications pour l'association.

    Les rapports de conformité indiquent si l'état de l'association est conforme ou non conforme, ainsi que le niveau de sévérité que vous spécifiez ici. Pour plus d'informations, consultez A propos de la conformité des associations State Manager.

    Le calendrier des modifications détermine l'instant d'exécution de l'association. Si le calendrier est fermé, l'association n'est pas appliquée. Si le calendrier est ouvert, l'association s'exécute en conséquence. Pour plus d'informations, consultez AWS Systems Manager Change Calendar.
12. Dans la section Rate control (Contrôle du rythme), sélectionnez les options permettant de contrôler la façon dont l'association s'exécute sur plusieurs nœuds gérés. 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.

    Dans la section Simultanéité, sélectionnez une option :

    - Sélectionnez targets (cibles) pour entrer un nombre absolu de cibles pouvant exécuter l'association simultanément.
    - Sélectionnez percentage (pourcentage) pour saisir un pourcentage de l'ensemble de cibles pouvant exécuter l'association simultanément.

    Dans la section Error threshold (Seuil d'erreurs), sélectionnez une option :

    - Sélectionnez errors (erreurs) pour saisir un nombre absolu d'erreurs autorisées avant que State Manager ne cesse d'exécuter des associations sur des cibles supplémentaires.
    - Sélectionnez percentage (pourcentage) pour saisir un pourcentage d'erreurs autorisées avant que State Manager ne cesse d'exécuter des associations sur des cibles supplémentaires.
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) Saisissez les noms de compartiment et de préfixe (dossier) dans les zones.

> **Note**
>
> Les autorisations S3 qui donnent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué au nœud géré, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, vérifiez que le profil d'instance ou le rôle de service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

Voici les autorisations minimales requises pour activer la sortie Amazon S3 pour une association. Vous pouvez restreindre davantage l'accès en attachant des politiques IAM à des utilisateurs ou à des rôles au sein d'un compte. Au minimum, un profil d'instance Amazon EC2 doit disposer d'un rôle IAM avec la politique gérée `AmazonSSMManagedInstanceCore` et la politique en ligne suivante.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:PutObjectAcl"
            ],
            "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
        }
    ]
}
```

Pour des autorisations minimales, le compartiment Amazon S3 vers lequel s'effectue l'exportation doit disposer des paramètres par défaut définis par la console Amazon S3. Pour plus d'informations sur la création de compartiments Amazon S3, consultez Création d'un compartiment dans le Guide de l'utilisateur Amazon S3.

> **Note**
>
> Les opérations d'API initiées par le document SSM lors d'une exécution d'association ne sont pas journalisées dans AWS CloudTrail.

14. Sélectionnez 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 AWS CLI (sous Linux ou Windows) ou les outils PowerShell pour créer une State Manager association. Cette section présente plusieurs exemples qui montrent comment utiliser les cibles et les contrôles du débit. Les cibles et les contrôles du rythme vous permettent d'affecter une association à des dizaines ou des centaines de nœuds, tout en contrôlant l'exécution de ces associations. 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.

Avant de commencer

Le paramètre `targets` est un tableau de critères de recherche qui cible les instances en utilisant une combinaison `Key`,`Value` (Clé-Valeur) que vous spécifiez. Si vous prévoyez de créer une association sur des dizaines ou des centaines de nœuds à l'aide du paramètre `targets`, passez en revue les options de ciblage suivantes avant de commencer la procédure.

Cibler des instances spécifiques en spécifiant des ID

```
--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

```
--targets Key=tag:tag-key,Values=tag-value-1,tag-value-2,tag-value-3
```

```
--targets Key=tag:Environment,Values=Development,Test,Pre-production
```

Ciblez des nœuds en utilisant AWS Resource Groups

```
--targets Key=resource-groups:Name,Values=resource-group-name
```

```
--targets Key=resource-groups:Name,Values=WindowsInstancesGroup
```

Ciblez toutes les instances dans les environnements actuels Compte AWS et Région AWS

```
--targets Key=InstanceIds,Values=*
```

> **Note**
>
> Notez les informations suivantes.
>
> - State Manager ne prend pas en charge l'exécution d'associations utilisant une nouvelle version d'un document si ce document est partagé à partir d'un autre compte. State Manager exécute toujours la version `default` d'un document s'il est partagé depuis un autre compte, même si la console Systems Manager indique qu'une nouvelle version a été traitée. Si vous souhaitez exécuter une association à l'aide d'une nouvelle version d'un document partagé à partir d'un autre compte, vous devez définir la version du document sur `default`.
> - Vous pouvez spécifier un maximum de cinq clés de balise en utilisant la AWS CLI. Si vous utilisez le AWS CLI, toutes les clés de balise spécifiées dans la `create-association` commande doivent être actuellement attribuées au nœud. Si elles ne le sont pas, State Manager ne parvient pas à cibler le nœud pour une association. Pour plus d'informations sur l'affectation d'identifications à vos nœuds, consultez Balisage des ressources Systems Manager.
> - Lorsque vous créez une association, vous spécifiez à quel moment le programme s'exécute. Spécifiez 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.

Créer une association

1. Installez et configurez le AWS CLI ou le AWS Tools for PowerShell, si ce n'est pas déjà fait.

   Pour plus d'informations, consultez la section Installation ou mise à jour de la version la plus récente de l' AWS CLI et Installation d' AWS Tools for PowerShell.

2. Utilisez le format suivant pour créer une commande qui crée une association State Manager. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   Linux & macOS

   ```
   aws ssm create-association \
       --name document_name \
       --document-version version_of_document_applied \
       --instance-id instances_to_apply_association_on \
       --parameters (if any) \
       --targets target_options \
       --schedule-expression "cron_or_rate_expression" \
       --apply-only-at-cron-interval required_parameter_for_schedule_offsets \
       --schedule-offset number_between_1_and_6 \
       --output-location s3_bucket_to_store_output_details \
       --association-name association_name \
       --max-errors a_number_of_errors_or_a_percentage_of_target_set \
       --max-concurrency a_number_of_instances_or_a_percentage_of_target_set \
       --compliance-severity severity_level \
       --calendar-names change_calendar_names \
       --target-locations aws_region_or_account \
       --tags "Key=tag_key,Value=tag_value"
   ```

   Windows

   ```
   aws ssm create-association ^
       --name document_name ^
       --document-version version_of_document_applied ^
       --instance-id instances_to_apply_association_on ^
       --parameters (if any) ^
       --targets target_options ^
       --schedule-expression "cron_or_rate_expression" ^
       --apply-only-at-cron-interval required_parameter_for_schedule_offsets ^
       --schedule-offset number_between_1_and_6 ^
       --output-location s3_bucket_to_store_output_details ^
   ```

```
    --association-name association_name ^
    --max-errors a_number_of_errors_or_a_percentage_of_target_set ^
    --max-concurrency a_number_of_instances_or_a_percentage_of_target_set ^
    --compliance-severity severity_level ^
    --calendar-names change_calendar_names ^
    --target-locations aws_region_or_account ^
    --tags "Key=tag_key,Value=tag_value"
```

PowerShell

```
New-SSMAssociation `
    -Name document_name `
    -DocumentVersion version_of_document_applied `
    -InstanceId instances_to_apply_association_on `
    -Parameters (if any) `
    -Target target_options `
    -ScheduleExpression "cron_or_rate_expression" `
    -ApplyOnlyAtCronInterval required_parameter_for_schedule_offsets `
    -ScheduleOffSet number_between_1_and_6 `
    -OutputLocation s3_bucket_to_store_output_details `
    -AssociationName association_name `
    -MaxError  a_number_of_errors_or_a_percentage_of_target_set
    -MaxConcurrency a_number_of_instances_or_a_percentage_of_target_set `
    -ComplianceSeverity severity_level `
    -CalendarNames change_calendar_names `
    -TargetLocations aws_region_or_account `
    -Tags "Key=tag_key,Value=tag_value"
```

L'exemple suivant crée une association sur des nœuds labélisés avec `"Environment,Linux"`. L'association utilise le document `AWS-UpdateSSMAgent` pour mettre à jour SSM Agent sur les nœuds ciblés à 2 h (UTC) chaque dimanche matin. Cette association s'exécute simultanément sur 10 nœuds maximum à un moment donné. En outre, cette association cesse d'être exécutée sur des nœuds 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 sévérité Medium (Moyenne).

Linux & macOS

```
aws ssm create-association \
  --association-name Update_SSM_Agent_Linux \
```

```
  --targets Key=tag:Environment,Values=Linux \
  --name AWS-UpdateSSMAgent  \
  --compliance-severity "MEDIUM" \
  --schedule-expression "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-expression "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"
    } `
  -ComplianceSeverity MEDIUM `
  -ScheduleExpression "cron(0 2 ? * SUN *)" `
  -MaxConcurrency 10 `
  -MaxError 5
```

L'exemple suivant cible les ID de nœuds en spécifiant une valeur générique (*). Cela permet à Systems Manager de créer une association sur tous les nœuds de l'actuel Compte AWS et Région AWS. Cette association s'exécute simultanément sur 10 nœuds maximum à un moment donné. En outre, cette association cesse d'être exécutée sur des nœuds 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 sévérité Medium (Moyenne). Cette association utilise un décalage de planification, ce qui signifie qu'elle s'exécute deux jours après la planification cron spécifiée. Elle inclut également le paramètre `ApplyOnlyAtCronInterval`,

qui est requis pour utiliser le décalage de planification, ce qui signifie que l'association ne sera pas exécutée immédiatement après sa création.

Linux & macOS

```
aws ssm create-association \
  --association-name Update_SSM_Agent_Linux \
  --name "AWS-UpdateSSMAgent" \
  --targets "Key=instanceids,Values=*" \
  --compliance-severity "MEDIUM" \
  --schedule-expression "cron(0 2 ? * SUN#2 *)" \
  --apply-only-at-cron-interval \
  --schedule-offset 2 \
  --max-errors "5" \
  --max-concurrency "10" \
```

Windows

```
aws ssm create-association ^
  --association-name Update_SSM_Agent_Linux ^
  --name "AWS-UpdateSSMAgent" ^
  --targets "Key=instanceids,Values=*" ^
  --compliance-severity "MEDIUM" ^
  --schedule-expression "cron(0 2 ? * SUN#2 *)" ^
  --apply-only-at-cron-interval ^
  --schedule-offset 2 ^
  --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#2 *)" `
  -ApplyOnlyAtCronInterval `
```

```
  -ScheduleOffset 2 `
  -MaxConcurrency 10 `
  -MaxError 5 `
  -ComplianceSeverity MEDIUM `
  -ApplyOnlyAtCronInterval
```

L'exemple suivant crée une association sur des nœuds dans des groupes de ressources. Le groupe est nommé « HR-Department ». L'association utilise le document `AWS-UpdateSSMAgent` pour mettre à jour SSM Agent sur les nœuds ciblés à 2 h (UTC) chaque dimanche matin. Cette association s'exécute simultanément sur 10 nœuds maximum à un moment donné. En outre, cette association cesse d'être exécutée sur des nœuds 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 sévérité Medium (Moyenne). Cette association s'exécute selon la planification Cron spécifiée. Il ne s'exécute pas immédiatement après la création de l'association.

Linux & macOS

```
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-expression "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-expression "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 nœuds balisés avec un ID de nœud spécifique. L'association utilise le document SSM Agent pour mettre à jour l'SSM Agent sur les nœuds ciblés une fois, lorsque le calendrier des modifications est ouvert. L'association vérifie l'état du calendrier lorsqu'elle s'exécute. Si le calendrier est fermé au moment du lancement et que l'association ne s'exécute qu'une seule fois, elle ne s'exécutera plus car la fenêtre d'exécution de l'association est passée. Si le calendrier est ouvert, l'association s'exécute en conséquence.

> **Note**
>
> Si vous ajoutez de nouveaux nœuds aux identifications ou aux groupes de ressources sur lesquels une association agit lorsque le calendrier des modifications est fermé, l'association est appliquée à ces nœuds une fois que le calendrier des modifications s'ouvre.

Linux & macOS

```
aws ssm create-association \
  --association-name CalendarAssociation \
  --targets "Key=instanceids,Values=i-0cb2b964d3e14fd9f" \
  --name AWS-UpdateSSMAgent  \
  --calendar-names "arn:aws:ssm:us-east-1:123456789012:document/testCalendar1" \
```

```
  --schedule-expression "rate(1day)"
```

Windows

```
aws ssm create-association ^
  --association-name CalendarAssociation ^
  --targets "Key=instanceids,Values=i-0cb2b964d3e14fd9f" ^
  --name AWS-UpdateSSMAgent  ^
  --calendar-names "arn:aws:ssm:us-east-1:123456789012:document/testCalendar1" ^
  --schedule-expression "rate(1day)"
```

PowerShell

```
New-SSMAssociation `
  -AssociationName CalendarAssociation `
  -Target @{
      "Key"="tag:instanceids"
      "Values"="i-0cb2b964d3e14fd9f"
    } `
  -Name AWS-UpdateSSMAgent `
  -CalendarNames "arn:aws:ssm:us-east-1:123456789012:document/testCalendar1" `
  -ScheduleExpression "rate(1day)"
```

L'exemple suivant crée une association qui s'exécute sur des nœuds balisés avec un ID de nœud spécifique. L'association utilise le document SSM Agent pour mettre à jour SSM Agent sur les nœuds ciblés à 2 h chaque dimanche matin. Cette association s'exécute uniquement selon la planification cron spécifiée lorsque le calendrier des modifications est ouvert. Lorsque l'association est créée, elle vérifie l'état du calendrier. Si le calendrier est fermé, l'association n'est pas appliquée. Lorsque l'intervalle d'application de l'association commence à 2h00 le dimanche, l'association vérifie si le calendrier est ouvert. Si le calendrier est ouvert, l'association s'exécute en conséquence.

> **Note**
>
> Si vous ajoutez de nouveaux nœuds aux identifications ou aux groupes de ressources sur lesquels une association agit lorsque le calendrier des modifications est fermé, l'association est appliquée à ces nœuds une fois que le calendrier des modifications s'ouvre.

Linux & macOS

```
aws ssm create-association \
  --association-name MultiCalendarAssociation \
  --targets "Key=instanceids,Values=i-0cb2b964d3e14fd9f" \
  --name AWS-UpdateSSMAgent  \
  --calendar-names "arn:aws:ssm:us-east-1:123456789012:document/testCalendar1"
 "arn:aws:ssm:us-east-2:123456789012:document/testCalendar2" \
  --schedule-expression "cron(0 2 ? * SUN *)"
```

Windows

```
aws ssm create-association ^
  --association-name MultiCalendarAssociation ^
  --targets "Key=instanceids,Values=i-0cb2b964d3e14fd9f" ^
  --name AWS-UpdateSSMAgent  ^
  --calendar-names "arn:aws:ssm:us-east-1:123456789012:document/testCalendar1"
 "arn:aws:ssm:us-east-2:123456789012:document/testCalendar2" ^
  --schedule-expression "cron(0 2 ? * SUN *)"
```

PowerShell

```
New-SSMAssociation `
  -AssociationName MultiCalendarAssociation `
  -Name AWS-UpdateSSMAgent `
  -Target @{
      "Key"="tag:instanceids"
      "Values"="i-0cb2b964d3e14fd9f"
    } `
  -CalendarNames "arn:aws:ssm:us-east-1:123456789012:document/testCalendar1"
 "arn:aws:ssm:us-east-2:123456789012:document/testCalendar2" `
  -ScheduleExpression "cron(0 2 ? * SUN *)"
```

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.

Modification et création de la nouvelle version d'une association

Vous pouvez modifier une association State Manager pour spécifier un nouveau nom, un niveau de sévérité ou des cibles. Vous pouvez aussi choisir d'écrire la sortie de la commande dans un compartiment Amazon Simple Storage Service (Amazon S3). Après avoir modifié une association, State Manager crée une nouvelle version. Vous pouvez afficher différentes versions après modification, comme décrit dans les procédures suivantes.

Les procédures suivantes décrivent comment modifier et créer une nouvelle version d'une association à l'aide de la console Systems Manager, AWS Command Line Interface (AWS CLI) et AWS Tools for PowerShell (Tools for PowerShell).

Important

State Manager ne prend pas en charge l'exécution d'associations utilisant une nouvelle version d'un document si ce document est partagé à partir d'un autre compte. State Manager exécute toujours la version `default` d'un document s'il est partagé à partir d'un autre compte, même si la console Systems Manager indique qu'une nouvelle version a été traitée. Si vous souhaitez exécuter une association à l'aide d'une nouvelle version d'un document partagé à partir d'un autre compte, vous devez définir la version du document sur `default`.

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 Amazon S3 existant. Si vous n'avez encore jamais utilisé d'Amazon S3, sachez que des frais s'appliquer à son utilisation. Pour plus d'informations sur la création d'un compartiment, consultez Créer un compartiment.

Pour modifier une association State Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez State Manager.
3. Sélectionnez l'association que vous avez créée dans Créer une association (ligne de commande), puis cliquez sur Edit (Modifier).
4. Dans le champ Name (Nom), saisissez un nouveau nom.
5. Dans la section Spécifier le programme, sélectionnez une nouvelle option.
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) Saisissez les noms de compartiment et de préfixe (dossier) dans les zones.

   > **Note**
   >
   > Les autorisations S3 qui donnent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué au nœud géré, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, vérifiez que le profil d'instance ou le rôle de service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

7. Sélectionnez Edit association. Configurez l'association en fonction de vos besoins actuels.
8. Dans la page Associations, sélectionnez le nom de l'association que vous venez de modifier, puis l'onglet Versions. Le système liste chaque version de l'association que vous avez créée et modifiée.
9. Ouvrez la console Amazon S3 sur https://console.aws.amazon.com/s3/.
10. Choisissez le nom du compartiment Simple Storage Service (Amazon S3) que vous avez spécifié pour le stockage de la sortie de commande, puis choisissez le dossier dont le nom est l'ID du nœud qui a exécuté l'association. (Si vous avez choisi de stocker la sortie dans un dossier du compartiment, ouvrez-le en premier.)
11. Explorez sur plusieurs niveaux dans le dossier `awsrunPowerShell` vers le fichier `stdout`.
12. Sélectionnez Open ou Download pour afficher le nom d'hôte.

Modifier une association (ligne de commande)

La procédure suivante décrit comment utiliser AWS CLI (sous Linux ou Windows) ou comment AWS Tools for PowerShell modifier et créer une nouvelle version d'une association.

Pour modifier une association State Manager

1. Installez et configurez le AWS CLI ou le AWS Tools for PowerShell, si ce n'est pas déjà fait.

   Pour plus d'informations, consultez la section Installation ou mise à jour de la version la plus récente de l' AWS CLI et Installation d' AWS Tools for PowerShell.

2. Utilisez le format suivant pour créer une commande permettant de modifier et de créer une nouvelle version d'une association State Manager existante. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   > **Important**
   >
   > Lorsque vous appelez `UpdateAssociation`, le système supprime tous les paramètres facultatifs de la demande et remplace l'association par des valeurs null pour ces paramètres. Ce comportement est intégré à la conception. Vous devez spécifier tous les paramètres facultatifs dans l'appel, même si vous ne modifiez pas les paramètres. Cela inclut le paramètre `Name`. Avant d'appeler cette action d'API, nous vous recommandons d'appeler l'opération d'DescribeAssociationAPI et de prendre note de tous les paramètres facultatifs requis pour votre `UpdateAssociation` appel.

   Linux & macOS

   ```
   aws ssm update-association \
       --name document_name \
       --document-version version_of_document_applied \
       --instance-id instances_to_apply_association_on \
       --parameters (if any) \
       --targets target_options \
       --schedule-expression "cron_or_rate_expression" \
       --schedule-offset "number_between_1_and_6" \
       --output-location s3_bucket_to_store_output_details \
       --association-name association_name \
       --max-errors a_number_of_errors_or_a_percentage_of_target_set \
       --max-concurrency a_number_of_instances_or_a_percentage_of_target_set \
   ```

```
    --compliance-severity severity_level \
    --calendar-names change_calendar_names \
    --target-locations aws_region_or_account
```

Windows

```
aws ssm update-association ^
    --name document_name ^
    --document-version version_of_document_applied ^
    --instance-id instances_to_apply_association_on ^
    --parameters (if any) ^
    --targets target_options ^
    --schedule-expression "cron_or_rate_expression" ^
    --schedule-offset "number_between_1_and_6" ^
    --output-location s3_bucket_to_store_output_details ^
    --association-name association_name ^
    --max-errors a_number_of_errors_or_a_percentage_of_target_set ^
    --max-concurrency a_number_of_instances_or_a_percentage_of_target_set ^
    --compliance-severity severity_level ^
    --calendar-names change_calendar_names ^
    --target-locations aws_region_or_account
```

PowerShell

```
Update-SSMAssociation `
    -Name document_name `
    -DocumentVersion version_of_document_applied `
    -InstanceId instances_to_apply_association_on `
    -Parameters (if any) `
    -Target target_options `
    -ScheduleExpression "cron_or_rate_expression" `
    -ScheduleOffset "number_between_1_and_6" `
    -OutputLocation s3_bucket_to_store_output_details `
    -AssociationName association_name `
    -MaxError  a_number_of_errors_or_a_percentage_of_target_set
    -MaxConcurrency a_number_of_instances_or_a_percentage_of_target_set `
    -ComplianceSeverity severity_level `
    -CalendarNames change_calendar_names `
    -TargetLocations aws_region_or_account
```

L'exemple suivant met à jour une association existante en remplaçant le nom par `TestHostnameAssociation2`. La nouvelle version d'association s'exécute toutes les heures et écrit la sortie des commandes dans le compartiment Amazon S3 spécifié.

Linux & macOS

```
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 en remplaçant le nom par `CalendarAssociation`. La nouvelle association s'exécute lorsque le calendrier est ouvert, et elle écrit la sortie de la commande dans le compartiment Amazon S3 spécifié.

Linux & macOS

```
aws ssm update-association \
  --association-id 8dfe3659-4309-493a-8755-01234EXAMPLE \
  --association-name CalendarAssociation \
  --parameters commands="echo Association" \
  --output-location S3Location='{OutputS3Region=us-
east-1,OutputS3BucketName=DOC-EXAMPLE-BUCKET,OutputS3KeyPrefix=logs}' \
  --calendar-names "arn:aws:ssm:us-east-1:123456789012:document/testCalendar2"
```

Windows

```
aws ssm update-association ^
  --association-id 8dfe3659-4309-493a-8755-01234EXAMPLE ^
  --association-name CalendarAssociation ^
  --parameters commands="echo Association" ^
  --output-location S3Location='{OutputS3Region=us-
east-1,OutputS3BucketName=DOC-EXAMPLE-BUCKET,OutputS3KeyPrefix=logs}' ^
  --calendar-names "arn:aws:ssm:us-east-1:123456789012:document/testCalendar2"
```

PowerShell

```
Update-SSMAssociation `
  -AssociationId b85ccafe-9f02-4812-9b81-01234EXAMPLE `
  -AssociationName CalendarAssociation `
  -AssociationName OneTimeAssociation `
  -Parameter @{"commands"="echo Association"} `
  -S3Location_OutputS3BucketName DOC-EXAMPLE-BUCKET `
  -CalendarNames "arn:aws:ssm:us-east-1:123456789012:document/testCalendar2"
```

L'exemple suivant met à jour une association existante en remplaçant le nom par `MultiCalendarAssociation`. La nouvelle association s'exécute lorsque les calendriers sont ouverts, et elle écrit la sortie de la commande dans le compartiment Amazon S3 spécifié.

Linux & macOS

```
aws ssm update-association \
  --association-id 8dfe3659-4309-493a-8755-01234EXAMPLE \
  --association-name MultiCalendarAssociation \
```

```
  --parameters commands="echo Association" \
  --output-location S3Location='{OutputS3Region=us-
east-1,OutputS3BucketName=DOC-EXAMPLE-BUCKET,OutputS3KeyPrefix=logs}' \
  --calendar-names "arn:aws:ssm:us-east-1:123456789012:document/testCalendar1"
 "arn:aws:ssm:us-east-2:123456789012:document/testCalendar2"
```

Windows

```
aws ssm update-association ^
  --association-id 8dfe3659-4309-493a-8755-01234EXAMPLE ^
  --association-name MultiCalendarAssociation ^
  --parameters commands="echo Association" ^
  --output-location S3Location='{OutputS3Region=us-
east-1,OutputS3BucketName=DOC-EXAMPLE-BUCKET,OutputS3KeyPrefix=logs}' ^
  --calendar-names "arn:aws:ssm:us-east-1:123456789012:document/testCalendar1"
 "arn:aws:ssm:us-east-2:123456789012:document/testCalendar2"
```

PowerShell

```
Update-SSMAssociation `
  -AssociationId b85ccafe-9f02-4812-9b81-01234EXAMPLE `
  -AssociationName MultiCalendarAssociation `
  -Parameter @{"commands"="echo Association"} `
  -S3Location_OutputS3BucketName DOC-EXAMPLE-BUCKET `
  -CalendarNames "arn:aws:ssm:us-east-1:123456789012:document/testCalendar1"
 "arn:aws:ssm:us-east-2:123456789012:document/testCalendar2"
```

3. Pour afficher la nouvelle version de l'association, exécutez la commande suivante.

Linux & macOS

```
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 retourne des informations telles que les suivantes.

Linux & macOS


```
{
    "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}
```

Suppression d'associations

La procédure suivante explique comment supprimer une State Manager association à l'aide de la AWS Systems Manager console.

Supprimer une association

Utilisez la procédure suivante pour supprimer une association à l'aide de la console AWS Systems Manager .

Pour supprimer une association

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez State Manager.
3. Sélectionnez une association, puis Supprimer.

Exécution de groupes Auto Scaling avec des associations

Une bonne pratique lors de l'utilisation d'associations pour exécuter des groupes Auto Scaling consiste à utiliser des cibles de balises. Si vous n'utilisez pas de balises, vous atteindrez peut-être la limite d'une association.

Si tous les nœuds sont labélisés avec la même clé et la même valeur, une seule association suffit pour exécuter votre groupe Auto Scaling. La procédure suivante décrit comment créer une telle association.

Pour créer une association qui exécute des groupes Auto Scaling

1. Vérifiez que tous les nœuds du groupe Auto Scaling sont bien labélisés avec la même clé et la même valeur. Pour obtenir des instructions sur l'étiquetage des nœuds, veuillez consulter Étiquetage de nœuds et de groupes Auto Scaling dans le Guide de l'utilisateur AWS Auto Scaling.
2. Créez une association à l'aide de la procédure dans Utilisation d'associations dans Systems Manager.

   Si vous utilisez la console, sélectionnez Specify instance tags (Spécifier des balises d'instance) dans le champ Targets (Cibles). Pour Instance tags (Balises d'instances), saisissez la clé Tag (Balise) et la valeur pour votre groupe Auto Scaling.

   Si vous utilisez la AWS Command Line Interface (AWS CLI), spécifiez `--targets Key=tag:tag-key,Values=tag-value` où la clé et la valeur correspondent à ce avec quoi vous avez labélisé vos nœuds.

Affichage des historiques des associations

Vous pouvez afficher toutes les exécutions correspondant à un ID d'association en utilisant l'opération d'API DescribeAssociationExecutions. Utilisez cette opération pour afficher le statut, le

statut détaillé, les résultats, l'heure de la dernière exécution, et d'autres informations relatives à une association State Manager. State Manager est une fonctionnalité de AWS Systems Manager. Cette opération d'API inclut également des filtres pour vous aider à trouver 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 afficher les exécutions qui ont été traitées après la date et l'heure spécifiées.

Si, par exemple, l'exécution d'une association a échoué, vous pouvez explorer en détail cette exécution en utilisant l'opération d'API DescribeAssociationExecutionTargets. Cette opération affiche les ressources, par exemple les ID de nœuds, dans lesquels l'association s'est exécutée et les divers statuts de l'association. Vous pouvez ensuite constater quel nœud ou quelle ressource n'est pas parvenu à exécuter une association. Vous pouvez utiliser l'ID de ressource pour afficher les détails de l'exécution d'une commande et identifier ainsi l'étape de la commande qui a échoué.

Les exemples contenus dans cette section incluent également des informations sur la manière d'utiliser l'opération d'API StartAssociationsOnce pour exécuter une association une seule fois au moment de sa création. Vous pouvez utiliser cette opération 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.

> **Note**
>
> Les opérations d'API initiées par le document SSM lors d'une exécution d'association ne sont pas journalisées dans AWS CloudTrail.

Affichage des historiques des associations (console)

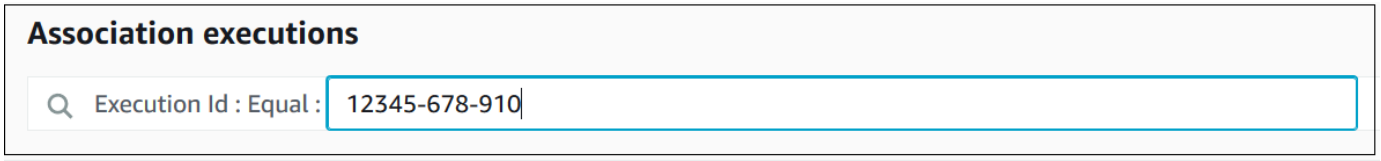
Utilisez la procédure suivante pour afficher l'historique d'association correspondant à un ID d'association spécifique et afficher ensuite les détails de l'exécution pour une ou plusieurs ressources.

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.

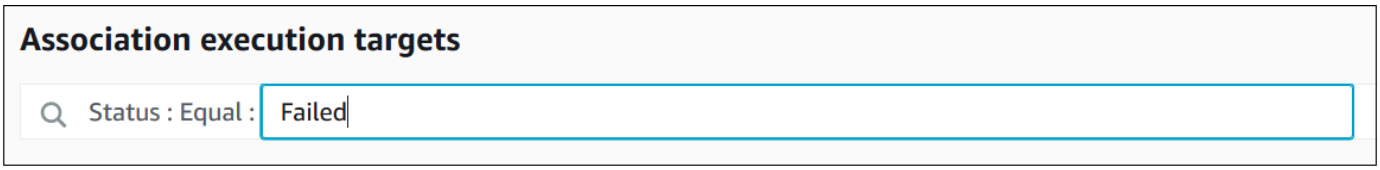
3. Dans le champ ID d'association, sélectionnez l'association dont vous souhaitez afficher l'historique.
4. Sélectionnez le bouton Afficher les détails.
5. Sélectionnez l'onglet Historique d'exécution.
6. Sélectionnez une association dont vous souhaitez afficher les détails d'exécution au niveau des ressources. Par exemple, sélectionnez une association qui indique le statut Échec. Vous pouvez ensuite consulter les détails d'exécution pour les nœuds qui n'ont pas réussi à exécuter l'association.

   Utilisez les filtres de la zone de recherche pour rechercher l'exécution dont vous souhaitez afficher les détails.



7. Sélectionnez 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. Sélectionnez un ID de ressource pour afficher les informations spécifiques relatives à cette ressource.

   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 une seule fois lors de sa création. Une fois que vous avez modifié la ressource sur laquelle l'exécution de l'association a échoué, sélectionnez le lien ID d'association dans le chemin de navigation.
10. Sélectionnez 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 Command Line Interface (AWS CLI) (sous Linux ou Windows) ou AWS Tools for 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 AWS Tools for PowerShell si vous ne l'avez pas déjà fait.

   Pour plus d'informations, consultez la section Installation ou mise à jour de la version la plus récente de l'AWS CLI et Installation d'AWS Tools for PowerShell.

2. Exécutez la commande suivante pour afficher la liste des exécutions correspondant à un ID d'association spécifique.

   Linux & macOS

   ```
   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"="GREAT`

Le système retourne des informations telles que les suivantes.

Linux & macOS

```
{
   "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 & macOS

```
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 & macOS

```
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 & macOS

   ```
   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 & macOS

```
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 un nœud géré spécifique sur lequel l'exécution d'une association a échoué.

Linux & macOS

```
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'opération d'API StartAssociationsOnce 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 & macOS

   ```
   aws ssm start-associations-once \
     --association-id ID
   ```

   Windows

   ```
   aws ssm start-associations-once ^
     --association-id ID
   ```

   PowerShell

   ```
   Start-SSMAssociationsOnce `
   ```

```
    -AssociationId ID
```

Utilisation d'associations avec IAM

State Manager, une fonctionnalité de AWS Systems Manager, utilise des cibles pour choisir les instances avec lesquelles vous configurez vos associations. À l'origine, les associations étaient créées en spécifiant un nom de document (`Name`) et un ID d'instance (`InstanceId`). Cela a créé une association entre un document et une instance ou un nœud géré. Auparavant, les associations étaient identifiées par ces paramètres. Bien que ces paramètres soient aujourd'hui obsolètes, ils sont toujours pris en charge. Les ressources `instance` et `managed-instance` ont été ajoutées en tant que ressources à des actions avec `Name` et `InstanceId`.

AWS Identity and Access Management Le comportement d'application des politiques (IAM) dépend du type de ressource spécifié. Les ressources pour des opérations State Manager sont appliquées uniquement sur la base de la demande transmise. State Manager ne vérifie pas en profondeur les propriétés des ressources de votre compte. Une demande n'est validée par rapport aux ressources de politique que si le paramètre de la demande contient les ressources de politique spécifiées. Par exemple, si vous spécifiez une instance dans le bloc de ressources, la politique est appliquée si la demande utilise le paramètre `InstanceId`. Pour chaque ressource du compte, le paramètre `InstanceId` n'est pas vérifié dans le paramètre `Targets`.

Voici quelques cas de comportement confus :

- DescribeAssociation, DeleteAssociation, et UpdateAssociationuse `instancemanaged-instance`, et `document` ressources pour spécifier la manière déconseillée de faire référence aux associations. Cela inclut toutes les associations créées avec le paramètre `InstanceId` obsolète.
- CreateAssociation, CreateAssociationBatch, ainsi que UpdateAssociationl'utilisation `instance` et `managed-instance` les ressources pour spécifier la manière déconseillée de faire référence aux associations. Cela inclut toutes les associations créées avec le paramètre `InstanceId` obsolète. Le type de ressource `document` fait partie de la manière obsolète de faire référence aux associations. C'est une propriété réelle d'une association. Cela signifie que vous pouvez créer des politiques IAM `Allow` ou des `Deny` autorisations pour les deux, `Create` ainsi que des `Update` actions basées sur le nom du document.

Pour de plus amples informations sur l'utilisation de politiques IAM avec Systems Manager, veuillez consulter Gestion des identités et des accès pour AWS Systems Manager ou Actions, ressources et clés de condition pour AWS Systems Manager dans la Référence des autorisations de service.

AWS Systems Manager State Manager Procédures

Les procédures suivantes démontrent comment créer et configurer les associations State Manager à l'aide de la console Systems Manager ou de l'AWS Command Line Interface (AWS CLI). Elles peuvent aussi vous montrer comment exécuter automatiquement les tâches administratives à l'aide de State Manager, une des fonctionnalités de AWS Systems Manager.

Rubriques

Démonstration : Création d'associations qui exécutent des fichiers MOF

Vous pouvez exécuter des fichiers MOF (Managed Object Format) pour appliquer l'état souhaité aux nœuds gérés par Windows Server dotés State Manager d'une fonctionnalité de AWS Systems Manager, à l'aide du 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 nœuds gérés sont 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 nœuds basés sur les ressources et leurs valeurs définies dans les fichiers MOF. Le document `AWS-ApplyDSCMofs` vous permet de télécharger et d'exécuter des fichiers de configuration MOF à partir d'Amazon Simple Storage Service (Amazon S3), d'un partage local ou 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 conformitéAWS Systems Manager.

L'exécution des fichiers MOF repose sur la configuration d'état PowerShell souhaitée (PowerShell DSC) de Windows. PowerShell DSC est une plate-forme déclarative utilisée pour la configuration, le déploiement et la gestion des systèmes Windows. PowerShell DSC permet aux administrateurs de décrire, dans de simples documents texte appelés configurations DSC, comment 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 peut être appliqué à un ou plusieurs serveurs afin d'obtenir la configuration souhaitée pour ces serveurs. PowerShell Les ressources DSC se chargent réellement de l'application de la configuration. Pour plus d'informations, consultez la section Présentation de la configuration de l'état PowerShell souhaité de Windows.

Rubriques

Utilisation d'Amazon S3 pour stocker les artefacts

Si vous utilisez Amazon S3 pour stocker des PowerShell modules, des fichiers MOF, des rapports de conformité ou des rapports d'état, le rôle AWS Identity and Access Management (IAM) utilisé par AWS Systems Manager SSM Agent doit avoir `GetObject` des `ListBucket` autorisations sur le compartiment. Si vous ne fournissez pas ces autorisations, le système renvoie une erreur Accès refusé. Voici des informations importantes sur le stockage d'artefacts dans Amazon S3.

- Si le compartiment se trouve dans un autre compartiment Compte AWS, créez une politique de ressources du compartiment qui accorde le compte (ou le rôle IAM) `GetObject` et `ListBucket` les autorisations.
- Si vous voulez utiliser des ressources DSC personnalisées, vous pouvez les télécharger à partir d'un compartiment Amazon S3. Vous pouvez également les installer automatiquement depuis la PowerShell galerie.
- Si vous utilisez Amazon S3 comme source de module, téléchargez le module sous forme de fichier Zip au format majuscules/minuscules suivant : *`ModuleName`*_ *`ModuleVersion`*.zip. Par 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. Cela vous permet de configurer la rotation automatique des informations d'identification. Cela permet également à DSC de propager automatiquement des informations d'identification vers vos serveurs sans redéployer des fichiers MOF.

Pour utiliser un AWS Secrets Manager secret dans une configuration, créez un objet PSCredential dont le nom d'utilisateur est le secret ou le SecretId SecretArn du secret contenant l'identifiant. 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
       }
    }
}
```

Compilez votre MOF en utilisant les PsAllowPlaintextPassword paramètres des 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 le nœud dispose d'un GetSecretValue accès dans le cadre d'une politique gérée par IAM, et éventuellement dans la politique de ressources secrètes s'il en existe une. 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-utilisateurs, dans laquelle vous avez deux noms d'utilisateur et mots de passe différents, et la AWS Lambda fonction de rotation alterne 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 ne fonctionne que pour les propriétés de ressource de type `String`. Toutefois, si votre ressource comporte une propriété de nœud CIM imbriquée, elle résout également les jetons CIM à partir des propriétés `String` dans ce nœud CIM. Vous ne pouvez pas utiliser la substitution de jeton pour des nombres ou des tableaux.

Par exemple, imaginez 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 tous vos nœuds. 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. Consultez l'exemple suivant.

```
Configuration MyConfig
{
    xComputer Computer
    {
        ComputerName = '{tag:ComputerName}'
    }
}
```

Vous définissez ensuite une identification sur le nœud géré dans la console Systems Manager, ou une identification Amazon Elastic Compute Cloud (Amazon EC2) dans la console Amazon EC2. Lorsque vous exécutez le document, le script remplace le jeton {tag :ComputerName} par la valeur de la balise d'instance.

Vous pouvez également combiner plusieurs balises dans une seule propriété, tel qu'illustré dans l'exemple suivant :

```
Configuration MyConfig
```

```
{
    File MyFile
    {
        DestinationPath = '{env:TMP}\{tag:ComputerName}'
        Type = 'Directory'
    }
}
```

Vous pouvez utiliser cinq types de jetons différents :

- identification : identifications Amazon EC2 ou de nœud géré.
- tagb64 : Identique à tag, mais le système utilise base64 pour décoder la valeur. Cela vous permet d'utiliser des caractères spéciaux dans des valeurs de balise.
- env : Résout des variables d'environnement.
- ssm : valeurs Parameter Store. Seuls les types String et Secure String sont pris en charge.
- tagssm : identique à tag, mais si l'identification n'est pas définie sur le nœud, le système tente de résoudre la valeur à partir d'un paramètre Systems Manager 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 un seul nœud (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, les délais d'allocation et les risques de dérive de configuration entre des nœuds regroupés à cause de l'installation de versions de module différentes sur le serveur de développement lors de la compilation de leurs fichiers MOF.

Prérequis

Avant de créer une association qui exécute des fichiers MOF, vérifiez que vos nœuds gérés ont les prérequis suivants installés :

- Windows PowerShell version 5.0 ou ultérieure. Pour plus d'informations, consultez la section Configuration PowerShell système requise pour Windows sur Microsoft.com.
- AWS Tools for Windows PowerShell version 3.3.261.0 ou ultérieure.
- SSM Agent 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 AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez State Manager.
3. Sélectionnez State Manager, puis Create association (Créer une association).
4. 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.
5. Dans la liste Document, sélectionnez **`AWS-ApplyDSCMofs`**.
6. Dans la section Parameters (Paramètres), spécifiez vos choix pour les paramètres d'entrée obligatoires et facultatifs.

   a. Mofs To Apply (Fichiers MOF à appliquer) : spécifiez un ou plusieurs fichiers MOF à exécuter lors de l'exécution de cette association. Utilisez des virgules pour séparer les fichiers MOF dans une liste. Vous pouvez spécifier les options suivantes pour localiser un fichier MOF.

   - 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 un 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. Service Path (Chemin d'accès au service) : (Facultatif) Un chemin d'accès au service est un préfixe d'un compartiment Amazon S3 dans lequel vous voulez 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 des paramètres, le système utilise {ssm:%servicePath%/*nom_paramètre* pour injecter la valeur de servicePath dans le nom de paramètre. *Par exemple, si le chemin de votre service est « WebServers /Production », le système résout le paramètre comme suit : WebServers /Production/ parameter_name.* Cela s'avère utile lorsque vous exécutez plusieurs environnements dans le même compte.

c. Report Bucket Name (Nom du compartiment des rapports) : (Facultatif) Saisissez le nom d'un compartiment Amazon S3 dans lequel vous voulez 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. Mof Operation Mode (Mode d'opération MOF) : sélectionnez le comportement State Manager lors de l'exécution de l'association **`AWS-ApplyDSCMofs`** :

- Apply (Appliquer) : corriger les configurations de nœuds qui ne sont pas conformes.
- ReportOnly: ne corrigez pas les configurations des nœuds, mais enregistrez toutes les données de conformité et signalez les nœuds non conformes.

e. Status Bucket Name (Statut du compartiment des rapports) : (Facultatif) Entrez le nom d'un compartiment Amazon S3 dans lequel vous voulez écrire les informations de statut d'exécution de fichier MOF. Ces rapports de statut sont des résumés de singleton de la dernière exécution de conformité d'un nœud. 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 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.

f. Nom du compartiment source du module : (Facultatif) Entrez le nom d'un compartiment Amazon S3 contenant les fichiers PowerShell du module. Si vous spécifiez None (Aucun), sélectionnez 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 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.

g. Autoriser la source du module PS Gallery : (Facultatif) Choisissez True pour télécharger PowerShell les modules depuis https://www.powershellgallery.com/. Si vous choisissez False, spécifiez une source pour l'option précédente, ModuleSourceBucketName.

h. Proxy Uri (URI du proxy) : (Facultatif) Utilisez cette option pour télécharger les fichiers MOF à partir d'un serveur proxy.

i. Reboot Behavior (Comportement de redémarrage) : (Facultatif) Spécifiez l'un des comportements de redémarrage suivants si votre exécution de fichier MOF nécessite un redémarrage :

- AfterMof: Redémarre le nœud une fois toutes les exécutions 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.
- Immediately (Immédiatement) : redémarre le nœud chaque fois qu'une exécution de fichier MOF le demande. Lors de l'exécution de plusieurs fichiers MOF demandant un redémarrage, le nœud est redémarré plusieurs fois.
- Never (Jamais) : les nœuds ne sont pas redémarrés, même si l'exécution de fichier MOF demande explicitement un redémarrage.

j. Use Computer Name For Reporting (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 des rapports de conformité. La valeur par défaut est false (faux), qui signifie que le système utilise l'ID de nœud lors de la création des rapports de conformité.

k. Enable Verbose Logging (Activer la journalisation détaillée) : (facultatif) nous vous recommandons d'activer la journalisation détaillée lors du déploiement de 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 des frais de stockage plus élevés pour Amazon S3. Pour éviter les problèmes de taille de stockage, nous vous recommandons d'activer des politiques de cycle de vie sur le compartiment Amazon S3. Pour de plus amples informations, consultez Comment créer une

politique de cycle de vie pour un compartiment S3 ? dans le Guide de l'utilisateur Amazon Simple Storage Service.

l. Enable Debug Logging (Activer la journalisation de débogage) : (facultatif) nous vous recommandons d'activer la journalisation de débogage pour 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 des frais de stockage plus élevés pour Amazon S3. Pour éviter les problèmes de taille de stockage, nous vous recommandons d'activer des politiques de cycle de vie sur le compartiment Amazon S3. Pour de plus amples informations, consultez Comment créer une politique de cycle de vie pour un compartiment S3 ? dans le Guide de l'utilisateur Amazon Simple Storage Service.

m. Compliance Type (Type de conformité) : (Facultatif) Spécifiez le type de conformité à utiliser pour la génération des rapports d'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 écrase les données de conformité existantes.

n. Pre Reboot Script (Script préalable au 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. Séparez les lignes supplémentaires par des points-virgules.

7. Dans la section Targets (Cibles), sélectionnez 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.

8. Dans la section Specify schedule (Spécifier le programme), sélectionnez On Schedule (Selon le calendrier) ou No schedule (Pas de calendrier). Si vous sélectionnez On Schedule (Selon

le calendrier), utilisez les boutons fournis pour créer un calendrier de type cron ou rate pour l'association.

9. Dans la section Advanced options (Options avancées) :

   - Dans Compliance severity (Sévérité de conformité), sélectionnez un niveau de sévérité pour l'association. Les rapports de conformité indiquent si l'état de l'association est conforme ou non conforme, ainsi que le niveau de sévérité que vous spécifiez ici. Pour plus d'informations, consultez A propos de la conformité des associations State Manager.

10. Dans la section Rate control (Contrôle de taux), configurez des options pour l'exécution d'associations State Manager dans un parc de nœuds gérés. Pour plus d'informations sur ces options, consultez À propos des cibles et des contrôles du débit dans les associations State Manager.

    Dans la section Simultanéité, sélectionnez une option :

    - Sélectionnez targets (cibles) pour entrer un nombre absolu de cibles pouvant exécuter l'association simultanément.
    - Sélectionnez percentage (pourcentage) pour saisir un pourcentage de l'ensemble de cibles pouvant exécuter l'association simultanément.

    Dans la section Error threshold (Seuil d'erreurs), sélectionnez une option :

    - Sélectionnez errors (erreurs) pour saisir un nombre absolu d'erreurs autorisées avant que State Manager cesse de d'exécuter des associations sur des cibles supplémentaires.
    - Sélectionnez percentage (pourcentage) pour saisir un pourcentage d'erreurs autorisées avant que State Manager cesse de d'exécuter des associations sur des cibles supplémentaires.

11. (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) Saisissez les noms de compartiment et de préfixe (dossier) dans les zones.

    > **Note**
    >
    > Les autorisations S3 qui donnent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué au nœud géré, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve

dans un autre compartiment Compte AWS, vérifiez que le profil d'instance ou le rôle de service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

12. Sélectionnez Create Association (Créer une association).

State Manager crée et exécute immédiatement l'association sur 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 :

- State Manager exécute des associations sur les nœuds qui sont en ligne lorsque l'intervalle commence, et ignore les nœuds hors ligne.
- State Manager tente d'exécuter l'association sur tous les nœuds configurés 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 de nœuds 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 consulter l'historique dans l'onglet Execution History (Historique d'exécution).

> **Note**
>
> Le `AWS-ApplyDSCMofs` est un document Command de Systems Manager. Cela signifie que vous pouvez également exécuter ce document en utilisant Run Command, une des fonctionnalités de AWS Systems Manager. Pour plus d'informations, consultez AWS Systems Manager Run Command.

Résolution des problèmes

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 :

1. Vérifiez que l'association est configurée pour écrire la sortie de commande dans Amazon S3 ou Amazon CloudWatch Logs (CloudWatch).
2. Définissez le paramètre Enable Verbose Logging (Activer la journalisation détaillée) sur True.
3. 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 nœuds, 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 :

- Le nœud comporte les autorisations d'accès requises à tous les compartiments Simple Storage Service (Amazon S3) liés aux fichiers MOF. En particulier :
  - s3 : GetObject autorisations : cela est obligatoire pour les fichiers MOF dans les compartiments Amazon S3 privés et les modules personnalisés dans les compartiments Amazon S3.
  - s3 : PutObject autorisation : cela est nécessaire pour écrire des rapports de conformité et l'état de conformité dans les compartiments Amazon S3.
- Si vous utilisez des identifications, assurez-vous que le nœud dispose des politiques IAM requises. L'utilisation de balises nécessite que le rôle IAM d'instance dispose d'une politique autorisant les actions `ec2:DescribeInstances` et `ssm:ListTagsForResource`.
- Assurez-vous que le nœud dispose des identifications 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 le nœud 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 nœud par rapport aux exigences de configuration figurant dans le fichier MOF. Pour voir le statut de conformité de vos nœuds, consultez les rapports de conformité. Vous pouvez afficher un rapport JSON dans le compartiment des rapports Amazon S3. Cela s'applique aux exécutions Run Command et State Manager. En outre, pour State Manager, vous pouvez afficher les détails de conformité sur la page de conformité Systems Manager.

Stderr spécifie : « Name resolution failure attempting to reach service » (Échec de résolution de nom lors d'une 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 a lieu lorsqu'un environnement informatique utilise un pare-feu ou un proxy transparent qui inclut une liste d'autorisations. 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:*`région-compartiment`*:*`nom-compartiment`*:*`nom-fichier-mof`*.mof.

  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 :

  `région-compartiment`:*`nom-compartiment`*. 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 le ou les nœuds ciblés peuvent accéder à Simple Storage Service (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, veuillez consulter la rubrique Points de terminaison de service Amazon S3 dans la Référence générale d'Amazon Web Services.

2. Connectez-vous au nœud cible et exécutez la commande ping suivante.

```
ping s3.s3-region.amazonaws.com
```

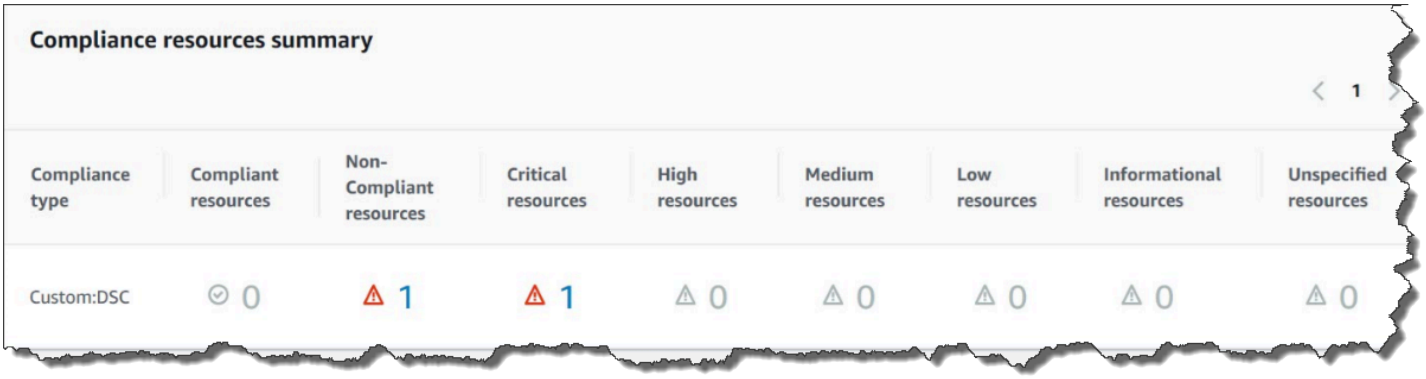
   Si le ping a échoué, cela signifie que Simple Storage Service (Amazon S3) est arrêté, qu'un pare-feu/proxy transparent bloque l'accès à la Région Simple Storage Service (Amazon S3), ou que le nœud ne peut pas accéder à Internet.

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 S3 que vous avez spécifié lorsque vous avez exécuté le document `AWS-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 ces informations sur la page Compliance (Conformité) de Systems Manager.

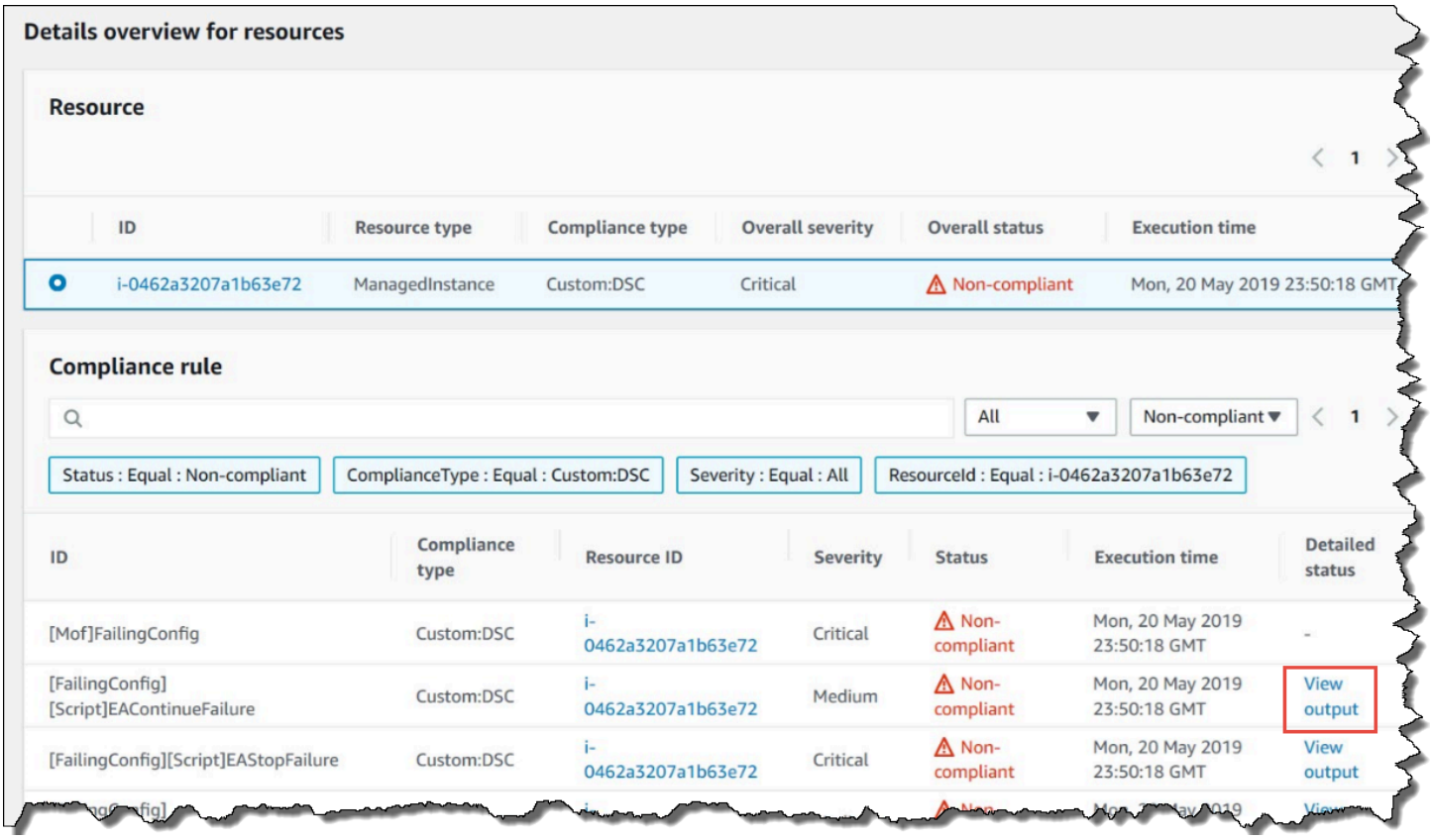
La section Récapitulatif des ressources de conformité affiche le nombre de ressources qui ont échoué. Dans l'exemple suivant, il ComplianceTypes'agit de Custom:DSC et une ressource n'est pas conforme.

> **Note**
>
> Custom:DSC est la ComplianceTypevaleur par défaut du document. `AWS-ApplyDSCMofs` Cette valeur est personnalisable.

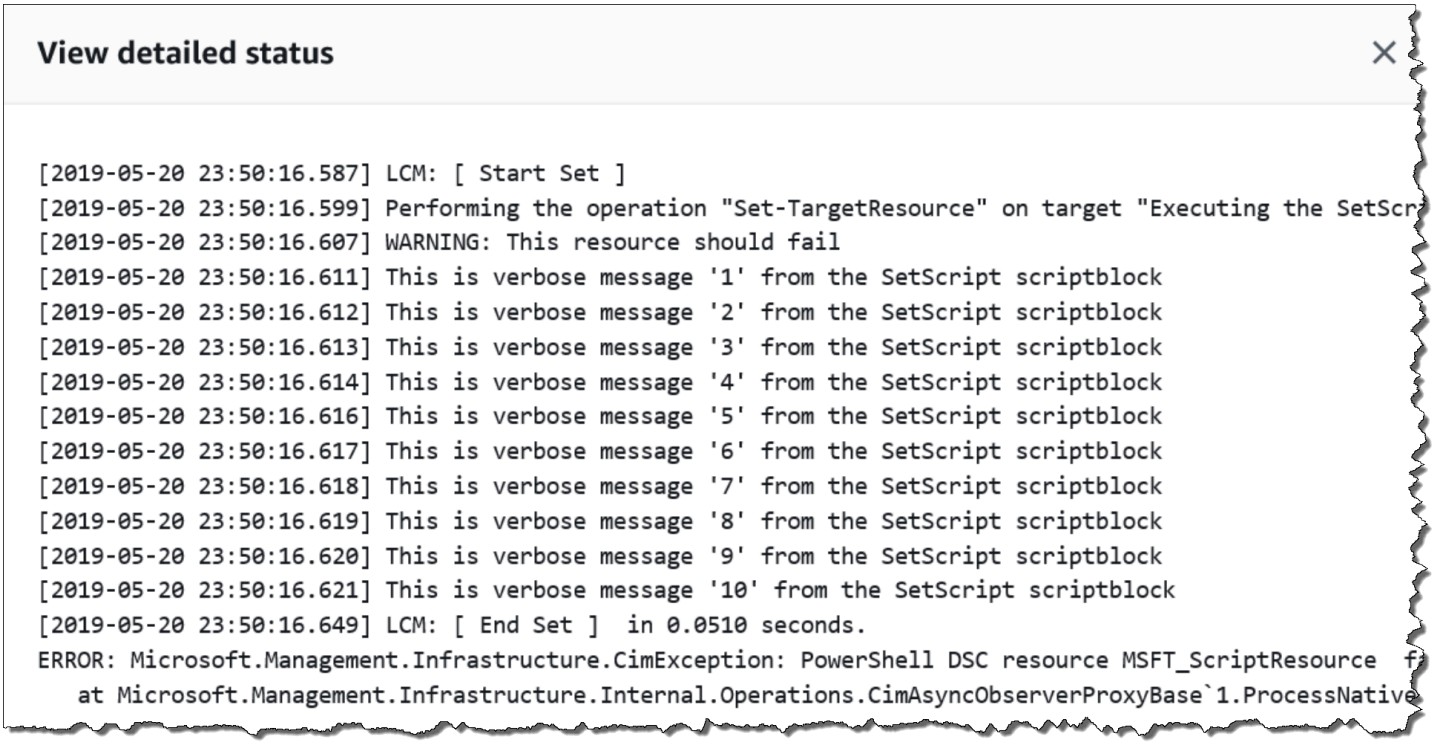


La section Présentation détaillée des ressources affiche des informations sur la AWS ressource contenant la ressource DSC non conforme. Cette section inclut également le nom MOF, les étapes

d'exécution du script et (le cas échéant) un lien View output (Afficher la sortie) pour consulter des informations de statut détaillées.



Le lien Afficher la sortie affiche les 4 000 derniers caractères du statut détaillé. Systems Manager utilise l'exception comme premier élément, puis analyse les messages détaillés à rebours, et en ajoute le plus possible jusqu'au quota 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, consultez Conformité d'AWS Systems Manager.

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 Simple Storage Service (Amazon S3) auquel le nœud 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 génère 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.

Procédure pas à pas : création d'associations qui exécutent Ansible des playbooks

Vous pouvez créer des State Manager associations qui exécutent Ansible des playbooks à l'aide du document `AWS-ApplyAnsiblePlaybooks` SSM. State Managerest une capacité de AWS Systems Manager. Ce document offre les avantages suivants pour l'exécution de manuels stratégiques :

- Prise en charge de l'exécution de manuels stratégiques complexes
- Support pour le téléchargement de playbooks depuis Amazon Simple Storage Service (Amazon S3) GitHub et depuis 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 Manager inclut deux documents SSM qui vous permettent de créer des State Manager associations qui exécutent des Ansible playbooks : `AWS-RunAnsiblePlaybook` et. `AWS-ApplyAnsiblePlaybooks` Le document `AWS-RunAnsiblePlaybook` 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.
> Les associations qui exécutent Ansible des playbooks ne sont pas prises en charge surmacOS.

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 ou le répertoire Zip peut être stocké dans GitHub 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 GitHub dans un seul fichier ou sous la forme d'un ensemble combiné de fichiers playbook. Pour télécharger du contenu depuisGitHub, spécifiez 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 playbooks à partir d'Amazon S3

Vous pouvez également stocker et télécharger Ansible des playbooks dans Amazon S3 sous forme de fichier .zip unique ou de structure de répertoire. Pour télécharger du contenu depuis Amazon S3, vous devez spécifier le chemin d'accès au fichier. Voici deux exemples :

Exemple 1 : Télécharger un fichier de manuel stratégique spécifique

```
{
    "path":"https://s3.amazonaws.com/DOC-EXAMPLE-BUCKET/playbook.yml"
}
```

Exemple 2 : Télécharger le 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 nœuds gérés doit être configuré avec la `AmazonS3ReadOnlyAccess` politique. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager.

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 spécifiéeAnsible.

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 correspondent directement aux options de Ansible verbosité.

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 InstallDependenciesparamètre, Systems Manager vérifie que les dépendances suivantes sont installées sur vos nœuds :

- Ubuntu Server/Debian Server: Apt-get (Gestion des paquets), Python 3, Unzip Ansible
- Amazon Linux : Ansible
- RHEL : Python 3Ansible, décompressez

Si une ou plusieurs de ces dépendances sont introuvables, Systems Manager les installe automatiquement.

Création d'une association qui exécute Ansible des playbooks (console)

La procédure suivante décrit comment utiliser la console Systems Manager pour créer une State Manager association qui exécute des Ansible playbooks à l'aide du `AWS-ApplyAnsiblePlaybooks` document.

Pour créer une association qui exécute des Ansible playbooks (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez State Manager.
3. Sélectionnez State Manager, puis Create association (Créer une association).
4. Pour Name (Nom), spécifiez un nom qui vous aide à mémoriser l'objectif de l'association.
5. Dans la liste Document, sélectionnez **`AWS-ApplyAnsiblePlaybooks`**.

6. Dans la section Paramètres, pour Type de source, choisissez S3 GitHubou S3.

   GitHub

   Si vous le souhaitez 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 sélectionnez S3, saisissez les informations de chemin au format suivant.

   ```
   {
      "path":"https://s3.amazonaws.com/path_to_directory_or_playbook_to_download"
   }
   ```

7. Pour Install Dependencies (Installer des dépendances), sélectionnez une option.
8. (Facultatif) Pour Playbook File (Fichier de manuel stratégique), entrez un nom de fichier. Si le playbook est contenu dans un fichier .zip, spécifiez un chemin d'accès relatif au fichier .zip.
9. (Facultatif) Pour Variables supplémentaires, entrez les variables auxquelles vous State Manager souhaitez envoyer au Ansible moment de l'exécution.
10. (Facultatif) Pour Check (Vérifier), sélectionnez une option.
11. (Facultatif) Pour Verbose (Détails), sélectionnez une option.
12. Pour Targets (Cibles), sélectionnez 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.
13. Dans la section Specify schedule (Spécifier une planification), sélectionnez On Schedule (Selon planification) ou No schedule (Pas de planification). Si vous sélectionnez On Schedule (Selon planification), utilisez les boutons fournis pour créer une planification de type cron ou rate pour l'association.
14. Dans la section Advanced options (Options avancées), pour Compliance severity (Sévérité de conformité), sélectionnez un niveau de sévérité pour l'association. Les rapports de conformité

indiquent si l'état de l'association est conforme ou non conforme, ainsi que le niveau de sévérité que vous spécifiez ici. Pour plus d'informations, consultez A propos de la conformité des associations State Manager.

15. Dans la section Rate control (Contrôle du débit), configurez des options pour l'exécution d'associations State Manager dans un parc de nœuds gérés. 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.

    Dans la section Simultanéité, sélectionnez une option :

    - Sélectionnez targets (cibles) pour entrer un nombre absolu de cibles pouvant exécuter l'association simultanément.
    - Sélectionnez percentage (pourcentage) pour saisir un pourcentage de l'ensemble de cibles pouvant exécuter l'association simultanément.

    Dans la section Error threshold (Seuil d'erreurs), sélectionnez une option :

    - Sélectionnez errors (erreurs) pour saisir un nombre absolu d'erreurs autorisées avant que State Manager ne cesse d'exécuter des associations sur des cibles supplémentaires.
    - Sélectionnez 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) Saisissez les noms de compartiment et de préfixe (dossier) dans les zones.

    > **Note**
    >
    > Les autorisations S3 qui donnent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué au nœud géré, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, vérifiez que le profil d'instance ou le rôle de service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

17. Sélectionnez Create Association (Créer une association).

> **Note**
>
> Si vous utilisez des identifications pour créer une association sur un ou plusieurs nœuds cibles, puis que vous supprimez les identifications d'un nœud, ce nœud n'exécute plus l'association. Le nœud est dissocié du document State Manager.

Créez une association qui exécute des Ansible playbooks (CLI)

La procédure suivante décrit comment utiliser le AWS Command Line Interface (AWS CLI) pour créer une State Manager association qui exécute des Ansible playbooks à l'aide du `AWS-ApplyAnsiblePlaybooks` document.

Pour créer une association qui exécute des Ansible playbooks (CLI)

1. Installez et configurez le AWS Command Line Interface (AWS CLI), si ce n'est pas déjà fait.

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l' AWS CLI.

2. Exécutez l'une des commandes suivantes pour créer une association qui exécute les Ansible playbooks en ciblant les nœuds à l'aide de balises. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations. La commande (A) indique GitHub le type de source. La commande (B) spécifie Amazon S3 comme type de source.

   (A) GitHub source

   Linux & macOS

```
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"],"TimeoutSeconds":["3600"]}' \
    --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"], "TimeoutSeconds":["3600"]}' ^
    --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 & macOS

```
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=Linux" \
    --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.

Le système tente de créer l'association sur les nœuds 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"
```

Procédure pas à pas : création d'associations qui exécutent Chef des recettes

Vous pouvez créer des State Manager associations qui exécutent Chef des recettes à l'aide du document `AWS-ApplyChefRecipes` SSM. State Managerest une capacité de AWS Systems Manager. Vous pouvez cibler des nœuds gérés par Systems Manager basés sur Linux avec le document SSM `AWS-ApplyChefRecipes`. Ce document offre les avantages suivants pour les Chef recettes de course à pied :

- Supporte plusieurs versions de Chef (Chef11 à Chef 18).
- Installe automatiquement le logiciel Chef client sur les nœuds cibles.
- Il exécute éventuellement des contrôles de conformité Systems Manager sur les nœuds cibles et stocke les résultats des contrôles de conformité dans un compartiment Amazon Simple Storage Service (Amazon 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 nœuds cibles sans y apporter de modifications.
- Il peut appliquer des attributs JSON personnalisés aux exécutions `chef-client`.
- Applique éventuellement des attributs JSON personnalisés à partir d'un fichier source stocké à l'emplacement spécifié.

Vous pouvez utiliser les compartiments Git GitHub, HTTP ou Amazon S3 comme sources de téléchargement pour les Chef livres de recettes et les recettes que vous spécifiez dans un `AWS-ApplyChefRecipes` document.

> **Note**
>
> Les associations qui exécutent Chef des recettes ne sont pas prises en charge surmacOS.

Conditions préalables : configurez votre association, votre référentiel et vos livres de recettes

Avant de créer un `AWS-ApplyChefRecipes` document, préparez vos livres de Chef recettes et votre référentiel de livres de recettes. Si vous n'avez pas encore de Chef livre de recettes que vous souhaitez utiliser, vous pouvez commencer par utiliser un `HelloWorld` livre de recettes de test préparé pour vous. 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` voici des exemples de Chef livres de cuisine disponibles Chef Supermarketsur le Chef site Web.

Bien que les livres de cuisine ne AWS puissent pas être officiellement pris en charge sur le Chef Supermarketsite Web, beaucoup d'entre eux fonctionnent avec le `AWS-ApplyChefRecipes` document. Voici des exemples de critères à déterminer 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 par Systems Manager que vous ciblez.
- Le livre de recettes doit être valide pour la version du Chef client (Chef11 à Chef 18) 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 `Chef.io` site Web 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 (document SSM). 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 des Chef recettes, sachez que le document exécuté installe le logiciel Chef client sur les nœuds gérés par Systems Manager, sauf si vous définissez la valeur de la version du Chef client sur. `None` Cette opération utilise un script d'installation provenant de Chef pour installer Chef les composants en votre nom. Avant de publier un `AWS-ApplyChefRecipes` document, assurez-vous que votre entreprise est en mesure de respecter toutes les exigences légales applicables, y compris les conditions

de licence applicables à l'utilisation des Chef logiciels. Pour plus d'informations, consultez le Chefsite Web.

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 nœuds gérés par 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 Systems Manager, consultez Conformité d'AWS Systems Manager.

Journalisation de l'exécution du document

Lorsque vous exécutez un document Systems Manager (document SSM) à l'aide d'une State Manager association, vous pouvez configurer l'association pour choisir la sortie du document à exécuter, et vous pouvez envoyer la sortie à Amazon S3 ou Amazon CloudWatch Logs (CloudWatch Logs). Pour faciliter le dépannage lorsqu'une association est terminée, vérifiez que l'association est configurée pour écrire la sortie de commande dans un compartiment Amazon S3 ou dans CloudWatch des journaux. Pour plus d'informations, consultez Utilisation d'associations dans Systems Manager.

Appliquer des attributs JSON aux cibles lors de l'exécution d'une recette

Vous pouvez spécifier les attributs JSON que votre Chef client doit appliquer aux nœuds cibles lors d'une exécution d'association. Lors de la configuration de l'association, vous pouvez fournir du code JSON brut ou indiquer le chemin d'accès à un fichier JSON stocké dans Amazon S3.

Utilisez les attributs JSON lorsque vous souhaitez personnaliser le mode d'exécution de la recette sans avoir à modifier la recette elle-même, par exemple :

- Remplacer un petit nombre d'attributs

  Utilisez JSON personnalisé pour éviter d'avoir à gérer plusieurs versions d'une recette pour tenir compte de différences mineures.
- Fournir des valeurs variables

  Utilisez le JSON personnalisé pour spécifier les valeurs susceptibles de changer par rapport à run-to-run. Par exemple, si vos Chef livres de recettes configurent une application tierce qui accepte

les paiements, vous pouvez utiliser un code JSON personnalisé pour spécifier l'URL du point de terminaison de paiement.

Spécifier des attributs en JSON brut

Voici un exemple du format que vous pouvez utiliser pour spécifier des attributs JSON personnalisés pour votre Chef recette.

```
{"filepath":"/tmp/example.txt", "content":"Hello, World!"}
```

Spécifier un chemin d'accès à un fichier JSON

Voici un exemple du format que vous pouvez utiliser pour spécifier le chemin d'accès aux attributs JSON personnalisés pour votre Chef recette.

```
{"sourceType":"s3", "sourceInfo":"someS3URL1"}, {"sourceType":"s3",
 "sourceInfo":"someS3URL2"}
```

Utiliser Git comme source de livre de recettes

Le `AWS-ApplyChefRecipes` document utilise le `aws:downloadContent`plugin pour télécharger des Chef livres de cuisine. Pour télécharger du contenu à partir de Git, spécifiez les informations relatives à votre référentiel Git au format JSON, comme dans l'exemple suivant. Remplacez chaque *`example-resource-placeholder`* avec vos propres informations.

```
{
   "repository":"GitCookbookRepository",
   "privateSSHKey":"{{ssm-secure:ssh-key-secure-string-parameter}}",
   "skipHostKeyChecking":"false",
   "getOptions":"branch:refs/head/main",
   "username":"{{ssm-secure:username-secure-string-parameter}}",
   "password":"{{ssm-secure:password-secure-string-parameter}}"
}
```

Utiliser GitHub comme source de livre de recettes

Le document `AWS-ApplyChefRecipes` utilise le plug-in `aws:downloadContent` pour télécharger les livres de recettes. Pour télécharger du contenu depuisGitHub, spécifiez les informations relatives à votre GitHub référentiel au format JSON, comme dans l'exemple suivant. Remplacez chaque *`example-resource-placeholder`* avec vos propres informations.

```
{
   "owner":"TestUser",
   "repository":"GitHubCookbookRepository",
   "path":"cookbooks/HelloWorld",
   "getOptions":"branch:refs/head/main",
   "tokenInfo":"{{ssm-secure:token-secure-string-parameter}}"
}
```

Utiliser HTTP comme source de livre de recettes

Vous pouvez stocker les Chef livres de recettes dans un emplacement HTTP personnalisé sous forme de `tar.gz` fichier unique `.zip` ou de structure de répertoire. Pour télécharger du contenu depuis HTTP, spécifiez le chemin d'accès au fichier ou au répertoire au format JSON comme dans l'exemple suivant. Remplacez chaque *example-resource-placeholder* avec vos propres informations.

```
{
   "url":"https://my.website.com/chef-cookbooks/HelloWorld.zip",
   "allowInsecureDownload":"false",
   "authMethod":"Basic",
   "username":"{{ssm-secure:username-secure-string-parameter}}",
   "password":"{{ssm-secure:password-secure-string-parameter}}"
}
```

Utiliser Amazon S3 comme source de livre de recettes

Vous pouvez également stocker et télécharger Chef des livres de recettes dans Amazon S3 sous forme de `tar.gz` fichier unique `.zip` ou de structure de répertoire. Pour télécharger du contenu depuis Amazon S3, spécifiez le chemin d'accès au fichier au format JSON comme dans les exemples suivants. Remplacez chaque *example-resource-placeholder* avec vos propres informations.

Exemple 1 : Télécharger un livre de recettes spécifique

```
{
   "path":"https://s3.amazonaws.com/chef-cookbooks/HelloWorld.zip"
}
```

Exemple 2 : Télécharger le 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 nœuds gérés doit être configuré avec la `AmazonS3ReadOnlyAccess` politique. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager.

Rubriques

- Création d'une association qui exécute Chef des recettes (console)
- Création d'une association qui exécute Chef des recettes (CLI)
- Affichage des détails de conformité des ressources Chef

Création d'une association qui exécute Chef des recettes (console)

La procédure suivante décrit comment utiliser la console Systems Manager pour créer une State Manager association qui exécute des Chef livres de recettes à l'aide du `AWS-ApplyChefRecipes` document.

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez State Manager.
3. Sélectionnez State Manager, puis Create association (Créer une association).
4. Pour Name (Nom), saisissez un nom qui vous aide à mémoriser l'objectif de l'association.
5. Dans la liste Document, sélectionnez **`AWS-ApplyChefRecipes`**.
6. Dans Paramètres, pour Type de source, sélectionnez Git GitHub, HTTP ou S3.
7. Pour Informations sur la source, entrez les informations sur la source du livre de recettes en utilisant le format approprié pour le Type de source que vous avez sélectionné à l'étape 6. Pour plus d'informations, consultez les rubriques suivantes :
   - the section called “Utiliser Git comme source de livre de recettes”
   - the section called “Utiliser GitHub comme source de livre de recettes”
   - the section called “Utiliser HTTP comme source de livre de recettes”

- the section called "Utiliser Amazon S3 comme source de livre de recettes"

8. 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. Remplacez chaque *example-resource-placeholder* avec vos propres informations.

```
recipe[cookbook-name1::recipe-name],recipe[cookbook-name2::recipe-name]
```

9. (Facultatif) Spécifiez les attributs JSON personnalisés que vous souhaitez que le Chef client transmette à vos nœuds cibles.

   a. Dans le contenu des attributs JSON, ajoutez les attributs que vous souhaitez que le Chef client transmette à vos nœuds cibles.

   b. Dans les sources d'attributs JSON, ajoutez les chemins d'accès aux attributs que vous souhaitez que le Chef client transmette à vos nœuds cibles.

   Pour plus d'informations, consultez the section called "Appliquer des attributs JSON aux cibles lors de l'exécution d'une recette".

10. Pour la version Chef client, spécifiez une Chef version. Les valeurs valides vont de `11` à `18` ou `None`. Si vous spécifiez un nombre compris entre `11` `18` (inclus), Systems Manager installe la bonne version du Chef client sur vos nœuds cibles. Si vous le spécifie`None`, Systems Manager n'installe pas le Chef client sur les nœuds cibles avant d'avoir exécuté les recettes du document.

11. (Facultatif) Pour les arguments Chef client, spécifiez des arguments supplémentaires pris en charge pour la version que Chef vous utilisez. Pour en savoir plus sur les arguments pris en charge, exécutez `chef-client -h` le programme sur un nœud qui exécute le Chef client.

12. (Facultatif) Activez Why-run pour afficher les modifications qui seront apportées aux nœuds cibles si les recettes sont exécutées, sans modifier réellement les nœuds cibles.

13. Pour Compliance severity (Sévérité de conformité), sélectionnez la sévérité des résultats de conformité de la configuration Systems Manager que vous voulez rapportés. Les rapports de conformité indiquent si l'état de l'association est conforme ou non conforme, ainsi que le niveau de sévérité 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 14). Pour de plus amples informations sur la conformité, consultez Utilisation du service Conformité dans ce guide.

Les scans de conformité mesurent le décalage entre la configuration spécifiée dans vos Chef recettes et les ressources des nœuds. Les valeurs valables sont `Critical`, `High`, `Medium`, `Low` `Informational` `Unspecified` ou `None`. Pour ignorer les rapports de conformité, sélectionnez `None`.

14. 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 associations State Manager ou `Custom:`*`custom-type`*. La valeur par défaut est `Custom:Chef`.
15. Pour Compliance report bucket, entrez le nom d'un bucket S3 dans lequel seront stockées les informations relatives à chaque Chef exécution effectuée par ce document, y compris la configuration des ressources et les résultats de conformité.
16. Dans Rate control (Contrôle du débit), configurez des options pour l'exécution d'associations State Manager dans un parc de nœuds gérés. 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.

    Dans Concurrency (Simultanéité), sélectionnez une option :

    - Sélectionnez targets (cibles) pour entrer un nombre absolu de cibles pouvant exécuter l'association simultanément.
    - Sélectionnez percentage (pourcentage) pour saisir un pourcentage de l'ensemble de cibles pouvant exécuter l'association simultanément.

    Dans Error threshold (Seuil d'erreur), sélectionnez une option :

    - Sélectionnez errors (erreurs) pour saisir un nombre absolu d'erreurs autorisées avant que State Manager ne cesse d'exécuter des associations sur des cibles supplémentaires.
    - Sélectionnez percentage (pourcentage) pour saisir un pourcentage d'erreurs autorisées avant que State Manager ne cesse d'exécuter des associations sur des cibles supplémentaires.
17. (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) Saisissez les noms de compartiment et de préfixe (dossier) dans les zones.

> **Note**
>
> Les autorisations S3 qui donnent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué au nœud géré, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, vérifiez que le profil d'instance ou le rôle de service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

18. Sélectionnez Create Association (Créer une association).

Création d'une association qui exécute Chef des recettes (CLI)

La procédure suivante décrit comment utiliser le AWS Command Line Interface (AWS CLI) pour créer une State Manager association qui exécute les livres de recettes Chef à l'aide du `AWS-ApplyChefRecipes` document.

1. Installez et configurez le AWS Command Line Interface (AWS CLI), si ce n'est pas déjà fait.

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l' AWS CLI.

2. Exécutez l'une des commandes suivantes pour créer une association qui exécute Chef des livres de recettes sur les nœuds cibles dotés des balises spécifiées. Utilisez la commande adaptée au type de source de votre livre de cuisine et à votre système d'exploitation. Remplacez chaque *`example-resource-placeholder`* avec vos propres informations.

   a. Source Git

      Linux & macOS

```
aws ssm create-association --name "AWS-ApplyChefRecipes" \
    --targets Key=tag:TagKey,Values=TagValue \
    --parameters '{"SourceType":["Git"],"SourceInfo":["{\"repository\":
\"repository-name\", \"getOptions\": \"branch:branch-name\", \"username
\": \"{{ ssm-secure:username-secure-string-parameter }}\", \"password\":
 \"{{ ssm-secure:password-secure-string-parameter }}\"}"], "RunList":
```

```
["{\"recipe[cookbook-name-1::recipe-name]\", \"recipe[cookbook-
name-2::recipe-name]\"}"], "JsonAttributesContent": ["{custom-json-
content}"], "JsonAttributesSources": "{\"sourceType\":\"s3\", \"sourceInfo
\":\"s3-bucket-endpoint-1\"}, {\"sourceType\":\"s3\", \"sourceInfo\":
\"s3-bucket-endpoint-2\"}", "ChefClientVersion": ["version-number"],
 "ChefClientArguments":["{chef-client-arguments}"], "WhyRun": boolean,
 "ComplianceSeverity": ["severity-value"], "ComplianceType":
 ["Custom:Chef"], "ComplianceReportBucket": ["s3-bucket-name"]}' \
    --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":["Git"],"SourceInfo":["{\"repository\":
\"repository-name\", \"getOptions\": \"branch:branch-name\", \"username
\": \"{{ ssm-secure:username-secure-string-parameter }}\", \"password\":
 \"{{ ssm-secure:password-secure-string-parameter }}\"}"], "RunList":
["{\"recipe[cookbook-name-1::recipe-name]\", \"recipe[cookbook-
name-2::recipe-name]\"}"], "JsonAttributesContent": ["{custom-json}"],
 "JsonAttributesSources": "{\"sourceType\":\"s3\", \"sourceInfo\":
\"s3-bucket-endpoint-1\"}, {\"sourceType\":\"s3\", \"sourceInfo\":
\"s3-bucket-endpoint-2\"}", "ChefClientVersion": ["version-number"],
 "ChefClientArguments":["{chef-client-arguments}"], "WhyRun": boolean,
 "ComplianceSeverity": ["severity-value"], "ComplianceType":
 ["Custom:Chef"], "ComplianceReportBucket": ["s3-bucket-name"]}' ^
    --association-name "name" ^
    --schedule-expression "cron-or-rate-expression"
```

b. GitHub source

Linux & macOS

```
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-name-1::recipe-name]\", \"recipe[cookbook-
name-2::recipe-name]\"}"], "JsonAttributesContent": ["{custom-json}"],
```

```
 "ChefClientVersion": ["version-number"], "ChefClientArguments":["{chef-
client-arguments}"], "WhyRun": boolean, "ComplianceSeverity": ["severity-
value"], "ComplianceType": ["Custom:Chef"], "ComplianceReportBucket": ["s3-
bucket-name"]}' \
    --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-name-1::recipe-name]\", \"recipe[cookbook-
name-2::recipe-name]\"}"], "JsonAttributesContent": ["{custom-json}"],
 "ChefClientVersion": ["version-number"], "ChefClientArguments":["{chef-
client-arguments}"], "WhyRun": boolean, "ComplianceSeverity": ["severity-
value"], "ComplianceType": ["Custom:Chef"], "ComplianceReportBucket": ["s3-
bucket-name"]}' ^
    --association-name "name" ^
    --schedule-expression "cron-or-rate-expression"
```

Voici un exemple.

Linux & macOS

```
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 *)"
```

Windows

```
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 *)"
```

c. Source HTTP

Linux & macOS

```
aws ssm create-association --name "AWS-ApplyChefRecipes" \
    --targets Key=tag:TagKey,Values=TagValue \
    --parameters '{"SourceType":["HTTP"],"SourceInfo":["{\"url\":\"url-
to-zip-file|directory|cookbook\", \"authMethod\": \"auth-method\",
 \"username\": \"{{ ssm-secure:username-secure-string-parameter }}\",
 \"password\": \"{{ ssm-secure:password-secure-string-parameter }}\"}"],
 "RunList":["{\"recipe[cookbook-name-1::recipe-name]\", \"recipe[cookbook-
name-2::recipe-name]\"}"], "JsonAttributesContent": ["{custom-json-
content}"], "JsonAttributesSources": "{\"sourceType\":\"s3\", \"sourceInfo
\":\"s3-bucket-endpoint-1\"}, {\"sourceType\":\"s3\", \"sourceInfo\":
\"s3-bucket-endpoint-2\"}", "ChefClientVersion": ["version-number"],
 "ChefClientArguments":["{chef-client-arguments}"], "WhyRun": boolean,
 "ComplianceSeverity": ["severity-value"], "ComplianceType":
 ["Custom:Chef"], "ComplianceReportBucket": ["s3-bucket-name"]}' \
    --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":["HTTP"],"SourceInfo":["{\"url\":\"url-
to-zip-file|directory|cookbook\", \"authMethod\": \"auth-method\",
 \"username\": \"{{ ssm-secure:username-secure-string-parameter }}\",
 \"password\": \"{{ ssm-secure:password-secure-string-parameter }}\"}"],
 "RunList":["{\"recipe[cookbook-name-1::recipe-name]\", \"recipe[cookbook-
name-2::recipe-name]\"}"], "JsonAttributesContent": ["{custom-json-
content}"], "JsonAttributesSources": "{\"sourceType\":\"s3\", \"sourceInfo
\":\"s3-bucket-endpoint-1\"}, {\"sourceType\":\"s3\", \"sourceInfo\":
\"s3-bucket-endpoint-2\"}", "ChefClientVersion": ["version-number"],
 "ChefClientArguments":["{chef-client-arguments}"], "WhyRun": boolean,
 "ComplianceSeverity": ["severity-value"], "ComplianceType":
 ["Custom:Chef"], "ComplianceReportBucket": ["s3-bucket-name"]}' \
    --association-name "name" ^
    --schedule-expression "cron-or-rate-expression"
```

d. Source Amazon S3

Linux & macOS

```
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.

Linux & macOS

```
aws ssm create-association --name "AWS-ApplyChefRecipes" \
    --targets "Key=tag:OS,Values= Linux" \
    --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 *)"
```

Windows

```
aws ssm create-association --name "AWS-ApplyChefRecipes" ^
    --targets "Key=tag:OS,Values= Linux" ^
    --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 *)"
```

Le système crée l'association et, à moins que votre expression cron ou rate spécifiée ne l'empêche, le système exécute l'association sur les nœuds cibles.

> **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.

3. Exécutez la commande suivante pour afficher le statut de l'association que vous venez de créer.

```
aws ssm describe-association --association-id "ID"
```

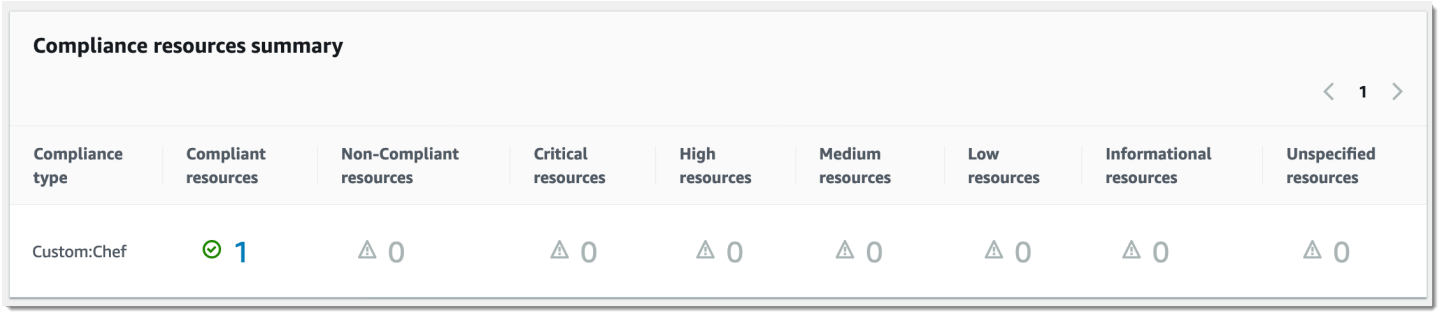
Affichage des détails de conformité des ressources Chef

Systems Manager capture les informations de conformité relatives aux ressources Chef gérées dans la valeur du compartiment du rapport de conformité Amazon S3 que vous avez spécifiée lors de l'exécution du `AWS-ApplyChefRecipes` document. La recherche d'informations sur les défaillances de Chef ressources dans un compartiment S3 peut prendre beaucoup de temps. Au lieu de cela, vous pouvez consulter ces informations sur la page Compliance (Conformité) de Systems Manager.

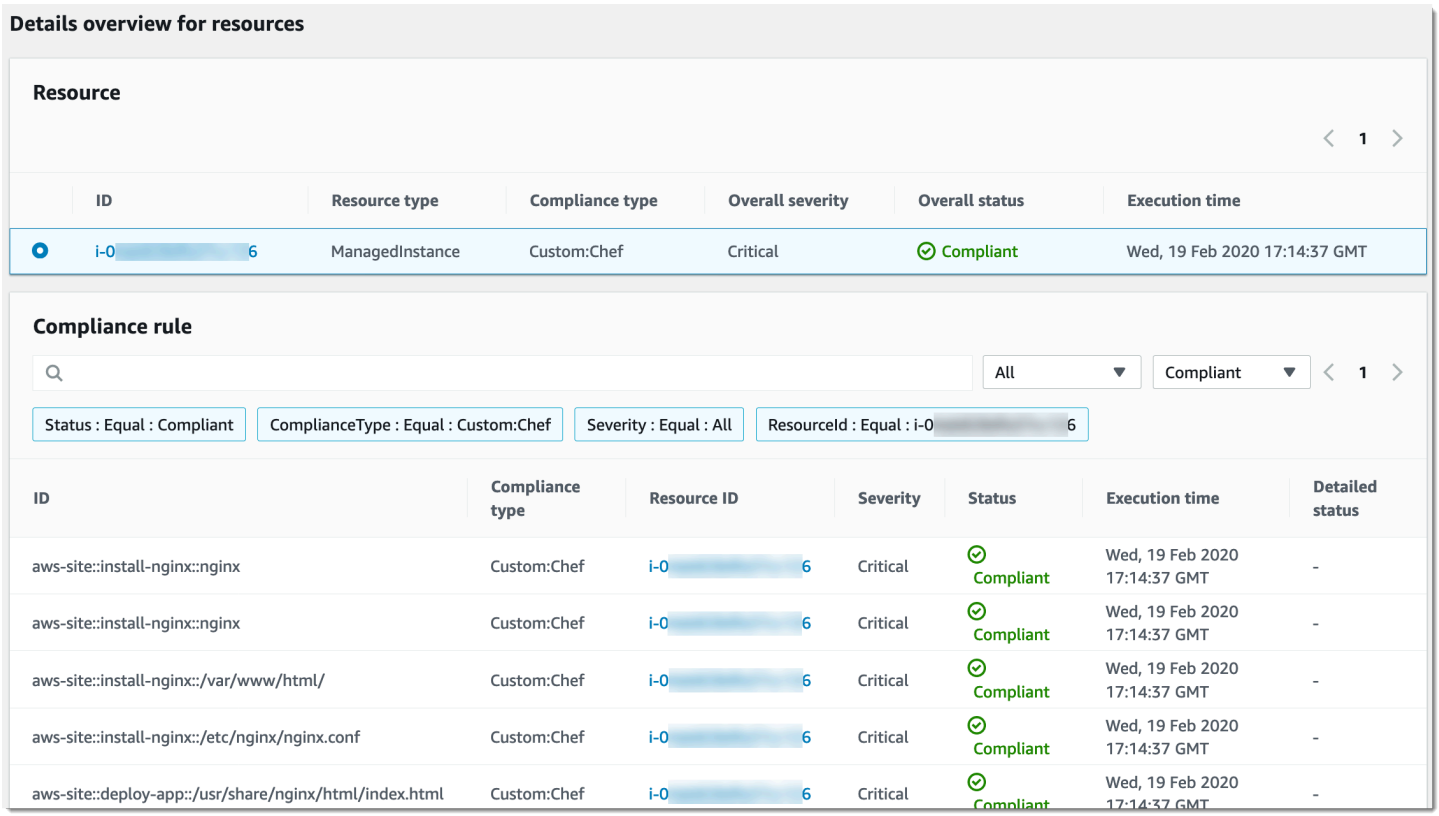
Une analyse de conformité de Systems Manager collecte des informations sur les ressources de vos nœuds gérés qui ont été créées ou vérifiées lors de la dernière Chef exécution. Les ressources peuvent inclure des fichiers, des répertoires, des services `systemd`, des packages `yum`, des fichiers modélisés, 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, il ComplianceTypes'agit de Custom : Chef et une ressource n'est pas conforme.

> **Note**
>
> `Custom:Chef`est la ComplianceTypevaleur par défaut du `AWS-ApplyChefRecipes` document. Cette valeur est personnalisable.



La section Présentation détaillée des ressources contient des informations sur la AWS ressource qui n'est pas conforme. Cette section inclut également le type de Chef ressource par rapport auquel la conformité a été exécutée, la gravité du problème, l'état de conformité et des liens vers des informations supplémentaires, le cas échéant.



Le lien Afficher la sortie affiche les 4 000 derniers caractères du statut détaillé. Systems Manager commence par l'exception comme premier élément, recherche les messages détaillés et les affiche jusqu'à ce qu'il atteigne le quota 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, consultez Conformité d'AWS Systems Manager.

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 Chef livre de recettes depuis un compartiment S3 auquel le nœud n'est pas autorisé à accéder, l'association échoue et Systems Manager ne fournit aucune donnée de conformité.

Démonstration : Mise à jour automatique de l'SSM Agent (CLI)

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. L'association met automatiquement à jour SSM Agent selon une planification que vous spécifiez. Pour plus d'informations sur SSM Agent, consultez Utilisation de l'option SSM Agent. Pour personnaliser le calendrier des mises à jour pour SSM Agent en utilisant la console, consultez Mise à jour automatique de l'SSM Agent.

Pour être informé des SSM Agent mises à jour, abonnez-vous à la page des notes de SSM Agent publication surGitHub.

Avant de commencer

Avant de réaliser la procédure suivante, veillez à avoir au moins une instance Amazon Elastic Compute Cloud (Amazon EC2) en cours d'exécution pour Linux, macOS ou Windows Server configurée pour Systems Manager. Pour plus d'informations, consultez Con AWS Systems Manager figuration.

Si vous créez une association en utilisant le AWS CLI ou AWS Tools for Windows PowerShell, utilisez le `--Targets` paramètre pour cibler les instances, comme illustré dans 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 SSM Agent

1. Installez et configurez le AWS Command Line Interface (AWS CLI), si ce n'est pas déjà fait.

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l' AWS CLI.
2. Exécutez la commande suivante pour créer une association en ciblant les instances avec des balises Amazon Elastic Compute Cloud (Amazon EC2). Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations.

Le paramètre `Schedule` définit une planification pour exécuter l'association tous les dimanches matin à 2 h 00 (UTC).

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.

Linux & macOS

```
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 *)"
```

Vous pouvez cibler plusieurs instances en spécifiant les ID d'instance dans une liste séparée par des virgules.

Linux & macOS

```
aws ssm create-association \
--targets Key=instanceids,Values=instance_ID,instance_ID,instance_ID \
--name AWS-UpdateSSMAgent \
--schedule-expression "cron(0 2 ? * SUN *)"
```

Windows

```
aws ssm create-association ^
--targets Key=instanceids,Values=instance_ID,instance_ID,instance_ID ^
--name AWS-UpdateSSMAgent ^
--schedule-expression "cron(0 2 ? * SUN *)"
```

Vous pouvez spécifier la version de l'SSM Agent à laquelle vous voulez faire la mise à jour.

Linux & macOS

```
aws ssm create-association \
--targets Key=instanceids,Values=instance_ID,instance_ID,instance_ID \
--name AWS-UpdateSSMAgent \
--schedule-expression "cron(0 2 ? * SUN *)" \
--parameters version=ssm_agent_version_number
```

Windows

```
aws ssm create-association ^
--targets Key=instanceids,Values=instance_ID,instance_ID,instance_ID ^
--name AWS-UpdateSSMAgent ^
--schedule-expression "cron(0 2 ? * SUN *)" ^
--parameters version=ssm_agent_version_number
```

Le système retourne des informations telles que les suivantes.

```
{
    "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 l'état la création d'état suivante. Le statut de l'association indique `Pending`.

3. Exécutez la commande suivante pour afficher le statut mis à jour de l'association que vous avez créé.

```
aws ssm list-associations
```

Si vos instances n'exécutent pas la version la plus récente de SSM Agent pour l'instant, l'état indique `Failed`. Lors de la publication d'une nouvelle version d'SSM Agent, l'association installe automatiquement le nouvel agent et le statut indique `Success`.

Procédure : Mettre à jour automatiquement les pilotes PV sur les instances EC2 pour Windows Server (console)

Les Amazon Machine Images (AMIs) Windows Amazon contiennent un jeu de pilotes permettant d'accéder au matériel virtualisé. Ces pilotes sont utilisés par Amazon Elastic Compute Cloud (Amazon EC2) pour mapper les volumes de stockage d'instance et 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 pas à pas suivante explique comment configurer une State Manager association pour télécharger et installer automatiquement de nouveaux pilotes AWS PV lorsque les pilotes sont disponibles. State Managerest une capacité de AWS Systems Manager.

Avant de commencer

Avant d'exécuter la procédure suivante, veillez à avoir au moins une instance Amazon EC2 pour Windows Server en cours d'exécution et configurée pour Systems Manager. Pour plus d'informations, consultez Con AWS Systems Manager figuration.

Pour créer une association State Manager qui met automatiquement à jour les pilotes PV

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.

2. Dans le panneau de navigation, sélectionnez State Manager.
3. Sélectionnez Créer une association.
4. Dans le champ Nom, entrez un nom descriptif pour l'association.
5. Dans la liste Document, sélectionnez `AWS-ConfigureAWSPackage`.
6. Dans la zone Paramètres, procédez comme suit :

   - Pour Actions, sélectionnez Installer.
   - Pour Installation type (Type d'installation), sélectionnez Uninstall and reinstall (Désinstaller et réinstaller).

   > **Note**
   >
   > Les mises à niveau sur place ne sont pas prises en charge pour ce package. Il doit être désinstallé et réinstallé.

   - Pour Name (Nom), saisissez **`AWSPVDriver`**.

     Il n'est pas nécessaire de saisir quoi que ce soit pour la version et les arguments supplémentaires.
7. Dans la section Targets (Cibles), sélectionnez les nœuds gérés sur lesquels vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant des instances ou des appareils de périphérie manuellement ou en spécifiant un groupe de ressources.

   > **Tip**
   >
   > Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés pour obtenir des conseils de dépannage.

   > **Note**
   >
   > Si vous choisissez de cibler les instances à l'aide de balises, et que vous spécifiez des balises qui correspondent à des instances Linux, l'association réussit sur l'instance Windows, mais échoue sur les instances Linux. Le statut global de l'association indique Failed.

8. Dans la zone Spécifier le calendrier, choisissez d'exécuter l'association selon un calendrier que vous configurez ou une seule fois. Comme des pilotes PV mis à jour sont seulement publiés quelques fois par an, vous pouvez planifier l'association pour qu'elle s'exécute une fois par mois, si vous le désirez.
9. Dans la zone Options avancées, pour le niveau de gravité de la 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 sévérité que vous spécifiez ici. Pour plus d'informations, consultez A propos de la conformité des associations State Manager.
10. Pour Rate control (Contrôle de débit) :
    - Dans Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage de nœuds gérés sur lesquels exécuter simultanément la commande.

      > **Note**
      >
      > Si vous avez sélectionné des cibles en spécifiant les balises appliquées aux nœuds gérés ou en spécifiant des groupes de ressources AWS , et que vous n'êtes pas certain du nombre de nœuds gérés ciblés, limitez le nombre de cibles autorisées à exécuter simultanément le document en indiquant un pourcentage.

    - Dans Error threshold (Seuil d'erreur), indiquez quand arrêter l'exécution de la commande sur les autres nœuds gérés après l'échec de celle-ci sur un certain nombre ou un certain pourcentage de nœuds. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les nœuds gérés sur lesquels la commande est toujours en cours de traitement peuvent également envoyer des erreurs.
11. (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) Saisissez les noms de compartiment et de préfixe (dossier) dans les zones.

    > **Note**
    >
    > Les autorisations S3 qui donnent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué au nœud géré, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, vérifiez que le profil d'instance ou le rôle de

service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

12. (Facultatif) Dans la section CloudWatch Alarme, pour Nom de l'alarme, choisissez une CloudWatch alarme à appliquer à votre association à des fins de surveillance.

> **Note**
>
> Notez les informations suivantes concernant ce passage.
>
> - La liste des alarmes affiche 100 alarmes maximum. Si votre alarme ne figure pas dans la liste, utilisez le AWS Command Line Interface pour créer l'association. Pour plus d'informations, consultez Créer une association (ligne de commande).
> - Pour associer une CloudWatch alarme à votre commande, le principal IAM qui crée l'association doit être autorisé à effectuer l'`iam:createServiceLinkedRole`action. Pour plus d'informations sur les CloudWatch alarmes, consultez la section Utilisation des CloudWatch alarmes Amazon.
> - Si votre alarme se déclenche, toute invocation ou automatisme de commande en attente ne s'exécute pas.

13. Sélectionnez Create association (Créer une association), puis Close (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 Amazon EC2 pour Windows Server, le statut se change en Success (Succès). 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 (Échec), sélectionnez l'ID de l'association, puis l'onglet Resources (Ressources) et vérifiez que l'association a été correctement créée sur vos instances EC2 pour Windows Server. Si les instances EC2 Windows Server affichent le statut Failed, vérifiez qu'elles SSM Agent sont en cours d'exécution sur l'instance et vérifiez que l'instance est configurée avec un rôle AWS Identity and Access Management (IAM) pour Systems Manager. Pour plus d'informations, voir Con AWS Systems Manager figuration.

AWS Systems Manager Patch Manager

Patch Manager, une fonctionnalité de AWS Systems Manager, automatise le processus d'application des correctifs aux nœuds gérés à la fois avec des mises à jour liées à la sécurité et d'autres types de mises à jour.

> **Important**
>
> À compter du 22 décembre 2022, Systems Manager prend en charge les politiques de correctifs, qui constituent la nouvelle méthode recommandée pour configurer vos opérations d'application de correctifs. À l'aide d'une configuration de politique de correctifs unique, vous pouvez définir l'application de correctifs pour tous les comptes de toutes les régions de votre organisation, uniquement pour les comptes et les régions de votre choix, ou pour une seule paire compte-région. Pour plus d'informations, consultez Utilisation des stratégies de correctifs Quick Setup.

Vous pouvez utiliser Patch Manager 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 publiées par Microsoft.) Vous pouvez utiliser Patch Manager pour installer des Service Packs sur les nœuds Windows et procéder à des mises à niveau mineures sur les nœuds Linux. Vous pouvez appliquer des correctifs aux flottes d'instances Amazon Elastic Compute Cloud (Amazon EC2), aux appareils de périphérie, aux serveurs sur site et aux machines virtuelles par type de système d'exploitation. Cela inclut les versions prises en charge de plusieurs systèmes d'exploitation, comme indiqué dans Conditions préalables requises Patch Manager. Vous pouvez scanner les instances pour afficher uniquement le rapport des correctifs manquants, ou scanner et installer automatiquement les correctifs manquants. Pour vos premiers pas dans Patch Manager, ouvrez Systems Manager console. Dans le panneau de navigation, sélectionnez Patch Manager.

> **Note**
>
> AWS ne teste pas les correctifs avant de les rendre disponibles dansPatch Manager. En outre, Patch Manager ne prend pas en charge la mise à niveau des principales versions des systèmes d'exploitation, telles que Windows Server 2016 à Windows Server 2019, ou SUSE Linux Enterprise Server (SLES) 12.0 à SLES 15.0.
>
> Pour les types de système d'exploitation basés sur Linux qui signalent un niveau de sévérité pour les correctifs, Patch Manager utilise le niveau de sévérité signalé par l'éditeur du logiciel pour l'avis de mise à jour ou le correctif individuel. Patch Manager ne dérive pas les niveaux

de sévérité de sources tierces, telles que le Common Vulnerability Scoring System (CVSS), ou des métriques publiées par la National Vulnerability Database (NVD).

Références de correctifs

Patch Manager utilise des référentiels de correctifs qui incluent les règles d'approbation automatique des correctifs quelques jours après leur publication, ainsi que des listes facultatives des correctifs approuvés et refusés. Lorsqu'une opération d'application de correctifs est exécutée, Patch Manager compare les correctifs actuellement appliqués à un nœud géré à ceux qui doivent être appliqués conformément aux règles définies dans le référentiel de correctifs. Vous pouvez faire en sorte que Patch Manager n'affiche qu'un rapport sur les correctifs manquants (une opération `Scan`), ou que Patch Manager installe automatiquement tous les correctifs manquants sur un nœud géré (une opération `Scan and install`).

Méthodes de fonctionnement d'application de correctifs

Patch Manager propose actuellement quatre méthodes d'exécution des opérations `Scan` et `Scan and install` :

- (Recommandé) Une politique de correctifs configurée dans Quick Setup — Sur la base de l'intégration avec AWS Organizations, une politique de correctifs unique peut définir des calendriers d'application des correctifs et des lignes de base de correctifs pour l'ensemble de l'organisation, y compris plusieurs comptes Comptes AWS et tous Régions AWS ceux sur lesquels ces comptes opèrent. Une politique de correctifs peut également cibler uniquement certaines unités d'organisation (UO) d'une organisation. Vous pouvez utiliser une seule politique de correctifs pour effectuer des analyses et des installations selon des planifications différentes. Pour plus d'informations, consultez Configuration des correctifs de l'organisation Patch Manager et Utilisation des stratégies de correctifs Quick Setup.
- Option de gestion des hôtes configurée dans Quick Setup : les configurations de gestion des hôtes sont également prises en charge par l'intégration à AWS Organizations, ce qui permet d'exécuter une opération d'application de correctifs pour l'ensemble d'une organisation au maximum. Toutefois, cette option se limite à rechercher les correctifs manquants à l'aide du référentiel de correctifs par défaut actuel et à fournir des résultats dans des rapports de conformité. Cette méthode de fonctionnement ne permet pas d'installer de correctifs. Pour plus d'informations, consultez Gestion des hôtes Amazon EC2.
- Fenêtre de maintenance pour exécuter une tâche de correctif **`Scan`** ou **`Install`** : une fenêtre de maintenance, que vous configurez dans la fonctionnalité Systems Manager appelée Maintenance

Windows, peut être configurée pour exécuter différents types de tâches selon une planification que vous définissez. Une tâche de type Run Command peut être utilisée pour exécuter des tâches `Scan` ou `Scan and install` sur un ensemble de nœuds gérés de votre choix. Chaque tâche de la fenêtre de maintenance peut cibler les nœuds gérés en une seule Compte AWSRégion AWS paire. Pour plus d'informations, consultez Démonstration : création d'une fenêtre de maintenance pour l'application des correctifs (console).

- Opération « Patch now » (Appliquer les correctifs maintenant) à la demande dans Patch Manager : l'option Patch now vous permet de contourner les configurations de planification lorsque vous devez appliquer des correctifs à des nœuds gérés le plus rapidement possible. À l'aide de Patch now, vous pouvez spécifier s'il faut exécuter l'opération `Scan` ou `Scan and install` et sur quels nœuds gérés l'exécuter. Vous pouvez également choisir d'exécuter les documents Systems Manager (documents SSM) en tant que crochets du cycle de vie pendant l'opération d'application des correctifs. Chaque opération Patch now peut cibler les nœuds gérés en une seule Compte AWSRégion AWS paire. Pour plus d'informations, consultez Application de correctifs sur les nœuds gérés à la demande.

Rapports de conformité

Après une opération `Scan`, vous pouvez utiliser la console Systems Manager pour afficher des informations sur les nœuds gérés qui ne sont pas conformes aux correctifs et sur les correctifs manquants sur chacun de ces nœuds. Vous pouvez également générer des rapports de conformité des correctifs au format .csv, qui sont envoyés à un compartiment Amazon Simple Storage Service (Amazon S3) de votre choix. Vous pouvez générer des rapports ponctuels ou selon un calendrier régulier. Pour un nœud géré individuel, les rapports contiennent les détails de tous les correctifs relatifs à ce nœud. Pour un ensemble de nœuds gérés, le rapport contient seulement un résumé indiquant le nombre de correctifs manquants. Une fois le rapport généré, vous pouvez utiliser un outil tel QuickSight qu'Amazon pour importer et analyser les données. Pour plus d'informations, consultez Utilisation des rapports de conformité des correctifs.

> **Note**
>
> Un élément de conformité généré par l'utilisation d'une politique de correctifs a le type d'exécution `PatchPolicy`. Un élément de conformité qui n'est pas généré lors d'une opération de politique de correctifs a le type d'exécution `Command`.

Intégrations

Patch Managers'intègre aux autres éléments suivants Services AWS :

- AWS Identity and Access Management (IAM) — Utilisez IAM pour contrôler quels utilisateurs, groupes et rôles ont accès aux Patch Manager opérations. Pour plus d'informations, consultez Fonctionnement d'AWS Systems Manager avec IAM et Configurer les autorisations d'instance requises pour Systems Manager.
- AWS CloudTrail— CloudTrail À utiliser pour enregistrer un historique vérifiable des événements liés aux opérations d'application de correctifs initiés par des utilisateurs, des rôles ou des groupes. Pour plus d'informations, consultez Journalisation des appels d' AWS Systems Manager API avec AWS CloudTrail.
- AWS Security Hub— Les données de conformité des correctifs Patch Manager peuvent être envoyées à AWS Security Hub. Security Hub vous offre une vue complète sur vos alertes de sécurité haute priorité et votre statut de conformité. Il surveille également le statut d'application des correctifs de votre flotte. Pour plus d'informations, consultez Intégration Patch Manager avec AWS Security Hub.
- AWS Config— Configurez l'enregistrement AWS Config pour afficher les données de gestion des instances Amazon EC2 dans le Patch Manager tableau de bord. Pour plus d'informations, voir Affichage des résumés du tableau de bord des correctifs.

Rubriques

Utilisation des stratégies de correctifs Quick Setup

À compter du 22 décembre 2022, Patch Manager propose une nouvelle méthode recommandée pour configurer l'application de correctifs pour votre organisation et Comptes AWS via l'utilisation de politiques relatives aux correctifs.

Une politique de correctifs est une configuration que vous définissez à l'aide de Quick Setup, une fonctionnalité d' AWS Systems Manager. Les politiques de correctif fournissent un contrôle plus étendu et plus centralisé de vos opérations d'application de correctifs qu'avec les méthodes précédentes. Les politiques de correctif peuvent être utilisées avec tous les systèmes d'exploitation pris en charge par Patch Manager, y compris les versions prises en charge de Linux, macOS et Windows Server. Pour plus d'informations sur la création de politiques de correctif, veuillez consulter la rubrique Configuration des correctifs de l'organisation Patch Manager.

Principales fonctionnalités des politiques de correctif

Au lieu d'utiliser d'autres méthodes pour appliquer des correctifs à vos nœuds, utilisez une politique de correctifs pour tirer parti des principales fonctionnalités suivantes :

- Configuration unique : la configuration des opérations d'application de correctifs à l'aide d'une fenêtre de maintenance ou d'une association State Manager peut requérir plusieurs tâches dans différentes parties de la console Systems Manager. À l'aide d'une politique de correctifs, toutes vos opérations d'application de correctifs peuvent être configurées dans un seul assistant.
- Support multicompte/multirégion : en utilisant une fenêtre de maintenance, une State Manager association ou la fonctionnalité Patch now dansPatch Manager, vous êtes limité à cibler les nœuds gérés en une seule paire. Compte AWS Région AWS Si vous utilisez plusieurs comptes et régions, vos tâches de configuration et de maintenance peuvent prendre beaucoup de temps, car vous devez effectuer des tâches de configuration dans chaque paire compte-région. Toutefois, si vous en utilisez AWS Organizations, vous pouvez configurer une politique de correctif qui s'applique à tous vos nœuds gérés dans l'ensemble de votre Comptes AWS. Régions AWS Ou, si vous le souhaitez, une politique de correctifs ne peut s'appliquer qu'à certaines unités organisationnelles (UO) des comptes et des régions de votre choix. Une politique de correctifs peut également s'appliquer à un seul compte local, si vous le souhaitez.
- Assistance à l'installation au niveau de l'organisation : l'option de configuration de gestion des hôtes existante dans Quick Setup permet d'effectuer une analyse quotidienne de la conformité aux correctifs de vos nœuds gérés. Toutefois, cette analyse est effectuée à une heure prédéterminée et n'aboutit qu'à des informations de conformité aux correctifs. Aucune installation de correctif n'est

effectuée. À l'aide d'une politique de correctifs, vous pouvez spécifier différentes planifications d'analyse et d'installation. Vous pouvez également choisir la fréquence et l'heure de ces opérations à l'aide d'expressions CRON ou Rate personnalisées. Par exemple, vous pouvez rechercher les correctifs manquants chaque jour afin de bénéficier d'informations de conformité régulièrement mises à jour. Toutefois, votre planification d'installation peut se limiter à une fois par semaine afin d'éviter des temps d'arrêt indésirables.

- Sélection simplifiée des référentiels de correctifs : les politiques de correctif intègrent toujours des référentiels de correctifs, et aucune modification n'a été apportée à la façon dont les référentiels de correctifs sont configurés. Toutefois, lorsque vous créez ou mettez à jour une politique de correctif, vous pouvez sélectionner la ligne de base AWS gérée ou personnalisée que vous souhaitez utiliser pour chaque type de système d'exploitation (OS) dans une liste unique. Il n'est pas nécessaire de spécifier le référentiel par défaut pour chaque type de système d'exploitation dans des tâches distinctes.

> **Note**
>
> Lors de l'exécution d'opérations d'application des correctifs basées sur une politique de correctifs, celles-ci utilisent le document SSM `AWS-RunPatchBaseline`. Pour de plus amples informations, veuillez consulter À propos du document SSM `AWS-RunPatchBaseline`.

Informations connexes

Déployez de manière centralisée les opérations de correction au sein de votre AWS organisation à l'aide de Systems Manager Quick Setup (blog sur les opérations et les migrations dans le AWS cloud)

Autres différences avec les politiques de correctif

Voici d'autres différences à noter lorsque vous utilisez des politiques de correctif au lieu des méthodes précédentes de configuration de l'application des correctifs :

- Aucun groupe de correctifs requis : lors des opérations d'application des correctifs précédentes, vous pouviez baliser plusieurs nœuds pour qu'ils appartiennent à un groupe de correctifs, puis spécifier le référentiel de correctifs à utiliser pour ce groupe de correctifs. Si aucun groupe de correctifs n'a été défini, Patch Manager a corrigé les instances avec le référentiel de correctifs par

défaut actuel pour le type de système d'exploitation. Grâce aux politiques de correctif, il n'est plus nécessaire de configurer et de gérer des groupes de correctifs.

- Page « Configurer l'application de correctifs » supprimée : avant la publication des politiques de correctif, vous pouviez spécifier des valeurs par défaut pour les nœuds à corriger, une planification d'application des correctifs et une opération d'application des correctifs sur la page Configure patching (Configurer l'application des correctifs). Cette page a été supprimée de Patch Manager. Ces options sont désormais spécifiées dans les politiques de correctif.
- Pas de support « Patch now » : la possibilité de patcher les nœuds à la demande est toujours limitée à une seule Compte AWS Région AWS paire à la fois. Pour plus d'informations, consultez Application de correctifs sur les nœuds gérés à la demande.
- Politiques de correctif et informations de conformité : lorsque vos nœuds gérés sont analysés à des fins de conformité, conformément à une configuration de politique d'application de correctifs, les données de conformité sont mises à votre disposition. Vous pouvez consulter et utiliser les données de la même manière qu'avec les autres méthodes d'analyse de conformité. Bien que vous puissiez configurer une politique de correctifs pour l'ensemble d'une organisation ou pour plusieurs unités organisationnelles, les informations de conformité sont signalées individuellement pour chaque Compte AWS Région AWS paire. Pour de plus amples informations, veuillez consulter Utilisation des rapports de conformité des correctifs.
- État de conformité des associations et politiques relatives aux correctifs : l'état des correctifs pour un nœud géré soumis à une politique de Quick Setup correctifs correspond au statut de l'exécution de l'State Managerassociation pour ce nœud. Si l'état d'exécution de l'association est`Compliant`, l'état d'application des correctifs pour le nœud géré est également marqué`Compliant`. Si l'état d'exécution de l'association est`Non-Compliant`, l'état d'application des correctifs pour le nœud géré est également marqué`Non-Compliant`.

Régions AWS pris en charge pour les politiques de correctifs

Actuellement, les configurations d'application de correctifs de Quick Setup sont prises en charge dans les régions suivantes :

- USA Est (Ohio) (us-east-2)
- USA Est (Virginie du Nord) (us-east-1)
- US Ouest (N. California) (us-west-1)
- USA Ouest (Oregon) (us-west-2)
- Asie-Pacifique (Mumbai) (ap-south-1)

- 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)

Conditions préalables requises Patch Manager

Assurez-vous de remplir les conditions requises avant d'utiliser Patch Manager une fonctionnalité de AWS Systems Manager.

Rubriques

Version de SSM Agent

La version 2.0.834.0 (ou version ultérieure) de SSM Agent doit être exécutée sur les nœuds gérés que vous souhaitez gérer avec Patch Manager.

> **Note**
>
> Une nouvelle version de SSM Agent est lancée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Le fait de ne pas utiliser la dernière version de l'agent peut empêcher votre nœud

géré d'utiliser diverses capacités et fonctionnalités de Systems Manager. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir SSM Agent à jour sur vos machines. Pour plus d'informations, consultez Automatisation des mises à jour de l'SSM Agent. Abonnez-vous à la page des notes de SSM Agent publication GitHub pour recevoir des notifications concernant les SSM Agent mises à jour.

Version Python

Pour macOS et la plupart des systèmes d'exploitation Linux, Patch Manager prend actuellement en charge les versions 2.6 à 3.10 de Python. Les systèmes d'exploitation AlmaLinux Debian ServerRaspberry Pi OS,, Ubuntu Server et nécessitent une version prise en charge de Python 3 (3.0 - 3.10).

Connectivité à la source de correctifs

Si vos nœuds gérés ne sont pas directement connectés à Internet et que vous utilisez une instance d'Amazon Virtual Private Cloud (Amazon VPC) avec un point de terminaison VPC, vous devez vous assurer que les nœuds ont accès aux référentiels de correctifs sources. Sur les nœuds Linux, les correctifs sont généralement téléchargés à partir des référentiels distants configurés sur le nœud. Par conséquent, le nœud doit être en mesure de se connecter aux référentiels afin que les correctifs puissent être appliqués. Pour plus d'informations, consultez Sélection des correctifs de sécurité.

Les nœuds gérés Windows Server doivent être en mesure de se connecter au catalogue Windows Update ou à Windows Server Update Services (WSUS). Vérifiez que vos nœuds sont connectés au catalogue Microsoft Update via une passerelle Internet, une instance NAT ou une passerelle NAT. Si vous utilisez WSUS, vérifiez que le nœud est connecté au serveur WSUS de votre environnement. Pour plus d'informations, consultez Problème : le nœud géré n'a pas accès au catalogue Windows Update ou à WSUS.

Accès aux points de terminaison S3

Que vos nœuds gérés fonctionnent sur un réseau privé ou public, sans accès aux compartiments Amazon Simple Storage Service (Amazon S3) AWS gérés requis, les opérations de correction échouent. Pour obtenir des informations sur les compartiments S3 auxquels vos nœuds gérés doivent avoir accès, consultez Communications de l'SSM Agent avec des compartiments S3 gérés par AWS et Améliorez la sécurité des instances EC2 en utilisant des points de terminaison VPC pour Systems Manager.

Systèmes d'exploitation pris en charge pour Patch Manager

Le Patch Manager ne prend pas en charge les mêmes versions des systèmes d'exploitation qui sont prises en charge par d'autres fonctionnalités de Systems Manager. Par exemple, Patch Manager ne prend pas en charge CentOS 6.3 ou Raspberry Pi OS OS 8 (Jessie). (Pour obtenir la liste complète des systèmes d'exploitation pris en charge par Systems Manager, consultez Systèmes d'exploitation pris en charge pour Systems Manager.) Par conséquent, assurez-vous que les nœuds gérés que vous souhaitez utiliser avec Patch Manager exécutent l'un des systèmes d'exploitation répertoriés dans le tableau suivant.

> **Note**
>
> Patch Managers'appuie sur les référentiels de correctifs configurés sur un nœud géré, tels que Windows Update Catalog et Windows Server Update Services pour Windows, pour récupérer les correctifs disponibles à installer. Par conséquent, pour les versions du système d'exploitation en fin de vie (EOL), si aucune nouvelle mise à jour n'est disponible, Patch Manager il est possible que vous ne puissiez pas signaler les nouvelles mises à jour. Cela peut être dû au fait qu'aucune nouvelle mise à jour n'est publiée par le responsable de la distribution Linux, Microsoft ou Apple, ou parce que le nœud géré ne dispose pas de la licence appropriée pour accéder aux nouvelles mises à jour.
> Patch Managerindique l'état de conformité par rapport aux correctifs disponibles sur le nœud géré. Par conséquent, si une instance exécute un système d'exploitation EOL et qu'aucune mise à jour n'est disponible, le nœud Patch Manager peut être signalé comme conforme, en fonction des lignes de base de correctifs configurées pour l'opération d'application de correctifs.

Système d'exploitation	Détails
Linux	• AlmaLinux 8,3—8,7, 9,0—9,2 • Amazon Linux 2012.03 à 2018.03 • Amazon Linux 2 version 2.0 et toutes les versions ultérieures • Amazon Linux 2022 • Amazon Linux 2023 • CentOS 6.5 à 7.9, 8.0 à 8.5

Système d'exploitation	Détails
	• CentOS Stream8 • Debian Server8.x, 9.x, 10.x, 11.x et 12.x • Oracle Linux 7.5 à 8.7, 9.0 à 9.2 • Raspberry Pi OS (anciennement Raspbian) 9 (Stretch) • Red Hat Enterprise Linux(RHEL) 6,5 à 8,9, 9,0 à 9,3 • Rocky Linux 8.4 à 8.7, 9.0 à 9.2 • SUSE Linux Enterprise Server(SLES) 12.0 et versions ultérieures 12. Versions X ; 15,0 à 15,5 • Ubuntu Server14,04 LTS, 16,04 LTS, 18,04 LTS, 20,04 LTS, 20,10 STR, 22,04 LTS et 23,04

Système d'exploitation	Détails
macOS	11.3.1; 11.4 à 11.7 (Big Sur) 12.0 à 12.6 (Monterey) 13.0 à 13.5 (Ventura) 14,0 (Sonoma) mises à jour du système d'exploitation macOS Patch Manager ne prend pas en charge les mises à jour ou les mises à niveau du système d'exploitation (SE) pour macOS, par exemple, de la version 12.x à la version 13.x ou de la version 13.1 à la version 13.2. Pour effectuer des mises à jour de la version du système d'exploitation sur macOS, nous vous recommandons d'utiliser les mécanisme s intégrés de mise à niveau du système d'exploitation d'Apple. Pour plus d'informations, consultez Gestion des appareils (français non garanti) sur le site web de la documentation du développeur Apple. Prise en charge de Homebrew Le système de gestion des packages logiciels open source Homebrew a cessé la prise en charge de macOS 10.14.x (Mojave) et 10.15.x (Catalina). Par conséquent, les opérations d'application de correctifs sur ces versions ne sont actuellement pas prises en charge. Prise en charge de la région macOSn'est pas pris en charge dans tous les cas Régions AWS. Pour plus d'informations sur

Système d’exploitation	Détails
	la prise en charge d'Amazon EC2 pourmacOS, consultez les instances Mac Amazon EC2 dans le guide de l'utilisateur Amazon EC2. Des appareils macOS Edge SSM Agentpour les appareils AWS IoT Greengrass principaux n'est pas pris en charge surmacOS. Vous ne pouvez pas utiliser Patch Manager pour appliquer des correctifs aux appareils de périphérie macOS.

Système d'exploitation	Détails
Windows	Windows Server 2008 à Windows Server 2022, y compris les versions R2. **Note** SSM Agentpour les appareils AWS IoT Greengrass principaux n'est pas pris en charge sous Windows 10. Vous ne pouvez pas utiliser Patch Manager pour appliquer des correctifs aux appareils de périphérie Windows 10. À propos de l'assistance Windows Server 2008 Depuis le 14 janvier 2020, Windows Server 2008 n'est plus pris en charge pour les mises à jour de fonctions ou de sécurité de Microsoft. Les Amazon Machine Images (AMIs) héritées, pour Windows Server 2008 et 2008 R2, incluent toujours la version 2 de l'SSM Agent, mais Systems Manager ne prend plus officiellement en charge les versions 2008 et ne met plus à jour l'agent pour ces versions de Windows Server. En outre, SSM Agent version 3 peut ne pas être compatibl e avec toutes les opérations sur Windows Server 2008 et 2008 R2. La version finale officiellement prise en charge de l'SSM Agent pour les versions Windows Server 2008 est 2.3.1644.0. À propos de la prise en charge de Windows Server 2012 et 2012 R2

Système d'exploitation	Détails
	Windows Server2012 et 2012 R2 ont atteint la fin du support le 10 octobre 2023. Pour utiliser Patch Manager ces versions, nous vous recommandons également d'utiliser les mises à jour de sécurité étendues (ESU) de Microsoft . Pour plus d'informations, consultez la section Windows Server2012 et 2012 R2 arrivant à la fin du support sur le site Web de Microsoft.

Fonctionnement des opérations Patch Manager

Cette section fournit des détails techniques sur la manière dont Patch Manager, une des fonctionnalités de AWS Systems Manager, détermine quels correctifs installer et dont 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, au sein d'un référentiel de correctifs personnalisé, pour les correctifs autres que ceux configurés par défaut sur un nœud géré. Cette section fournit également des détails sur le fonctionnement des règles de référentiel de correctifs sur différentes distributions du système d'exploitation Linux.

> **Note**
>
> Les informations des rubriques suivantes s'appliquent, quels que soient la méthode ou le type de configuration que vous utilisez pour vos opérations d'application de correctifs :
>
> - Une politique de correctifs configurée dans Quick Setup
> - Une option de gestion des hôtes configurée dans Quick Setup
> - Une fenêtre de maintenance pour exécuter un correctif `Scan` ou une tâche `Install`
> - Une opération Patch now (Appliquer les correctifs maintenant) à la demande

Rubriques

- Installation des correctifs
- Fonctionnement des règles de référence de correctif sur les systèmes basés sur Linux
- Différences clés entre l'application de correctifs dans Windows et Linux

Calcul des dates de sortie et des mises à jour des packages

> **Important**
>
> Les informations de cette page s'appliquent aux systèmes d'exploitation (OS) Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022 et Amazon Linux 2023 pour les instances Amazon Elastic Compute Cloud (Amazon EC2). Amazon Web Services crée et gère les packages pour ces types de systèmes d'exploitation. La gestion des packages et des référentiels par les fabricants d'autres systèmes d'exploitation influe sur le calcul de leurs dates de sortie et de mise à jour. Pour les systèmes d'exploitation autres qu'Amazon Linux, Amazon Linux 2, Amazon Linux 2022 et Amazon Linux 2023, tels que Red Hat Enterprise Linux (RHEL) et SUSE Linux Enterprise Server (SLES), consultez la documentation du fabricant pour plus d'informations sur la mise à jour et l'entretien de ses packages.

Dans les paramètres des lignes de base de correctifs personnalisées que vous créez, pour la plupart des types de systèmes d'exploitation, vous pouvez spécifier que l'installation des correctifs est approuvée automatiquement après un certain nombre de jours. AWS fournit plusieurs lignes de base de correctifs prédéfinies qui incluent des dates d'approbation automatique de 7 jours.

Un délai d'auto-approbation correspond au nombre de jours à attendre après la publication du correctif et avant son approbation. Par exemple, vous créez une règle à l'aide de la classification `CriticalUpdates` et la configurez pour un délai d'approbation automatique de 7 jours. Par conséquent, pour un nouveau correctif critique dont la date de sortie ou de la dernière mise à jour est le 7 juillet, l'approbation automatique aura lieu le 14 juillet.

Pour éviter les résultats inattendus liés aux retards d'approbation automatique sur Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022 et Amazon Linux 2023, il est important de comprendre comment leurs dates de sortie et de mise à jour sont calculées.

Dans la plupart des cas, le temps d'attente de l'approbation automatique avant l'installation des correctifs est calculé à partir d'une valeur `Updated Date` dans `updateinfo.xml`, et non pas d'une valeur `Release Date`. Voici des détails importants relatifs à ces calculs de date :

- La `Release Date` est la date à laquelle un avis est publié. Cette publication ne signifie pas forcément que le package est disponible dans les référentiels associés.
- La `Update Date` est la dernière date de la mise à jour de l'avis. La mise à jour d'un avis peut être aussi petite que la mise à jour d'un texte ou d'une description. Cette mise à jour ne signifie pas que le package est publié à partir de cette date ni qu'il est forcément disponible dans les référentiels associés.

  Cela signifie qu'un package peut avoir une valeur `Update Date` du 7 juillet mais être indisponible pour l'installation avant (par exemple) le 13 juillet. Supposons dans ce cas qu'un référentiel de correctifs spécifiant un délai d'approbation automatique de 7 jours s'exécute dans une opération `Install` le 14 juillet. Étant donné que la valeur `Update Date` est de 7 jours avant la date d'exécution, les correctifs et les mises à jour du package sont installés le 14 juillet. L'installation a lieu même si un seul jour s'est écoulé depuis que le package est disponible pour une installation réelle.
- Après sa publication initiale, un package contenant des correctifs de système d'exploitation ou d'application peut être mis à jour plusieurs fois.
- Un package peut être publié dans les référentiels AWS gérés, puis annulé si des problèmes sont découverts ultérieurement.

Dans certaines opérations d'application de correctifs, ces facteurs peuvent être insignifiants. Par exemple, si la configuration d'un référentiel de correctifs permet l'installation d'un correctif dont les valeurs de gravité sont de `Low` et `Medium`, et une classification de `Recommended`, tout retard de l'approbation automatique aura peu d'impact sur vos opérations.

Toutefois, dans les cas où la synchronisation des correctifs critiques ou de gravité élevée est plus importante, vous pouvez exercer un plus grand contrôle lors de l'installation des correctifs. La méthode recommandée pour ce faire consiste à utiliser des référentiels sources de correctifs alternatifs au lieu des référentiels par défaut pour les opérations d'application de correctifs sur un nœud géré.

Vous pouvez spécifier d'autres référentiels source de correctifs lors de la création d'un référentiel de correctifs personnalisée. Dans chaque référentiel 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. Pour de plus amples informations, veuillez consulter Spécification d'un autre référentiel source de correctifs (Linux).

Sélection des correctifs de sécurité

L'objectif principal de la fonctionnalité Patch Manager d' AWS Systems Manager est d'installer les mises à jour liées à la sécurité des systèmes d'exploitation sur les nœuds gérés. Par défaut, Patch Manager n'installe pas tous les correctifs disponibles, mais plutôt un plus petit ensemble de correctifs axé sur la sécurité.

Pour les types de système d'exploitation basés sur Linux qui signalent un niveau de sévérité pour les correctifs, Patch Manager utilise le niveau de sévérité signalé par l'éditeur du logiciel pour l'avis de mise à jour ou le correctif individuel. Patch Manager ne dérive pas les niveaux de sévérité de sources tierces, telles que le Common Vulnerability Scoring System (CVSS), ou des métriques publiées par la National Vulnerability Database (NVD).

> **Note**
>
> Sur tous les systèmes Linux pris en charge par Patch Manager, vous pouvez choisir un autre référentiel source configuré pour le nœud géré, généralement pour installer des mises à jour non liées à la sécurité. Pour plus d'informations, veuillez consulter Spécification d'un autre référentiel source de correctifs (Linux).

Le reste de cette section explique comment Patch Manager sélectionne les correctifs de sécurité pour les différents systèmes d'exploitation pris en charge.

Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022, and Amazon Linux 2023

Les référentiels préconfigurés sont gérés différemment sur Amazon Linux 1 et Amazon Linux 2 par rapport à Amazon Linux 2022 et Amazon Linux 2023.

Sur Amazon Linux 1 et Amazon Linux 2, le service de base de correctifs de Systems Manager utilise des référentiels préconfigurés sur le nœud géré. Un nœud comporte généralement deux référentiels préconfigurés :

Sur Amazon Linux 1

- ID de référentiel : `amzn-main/latest`

  Nom de référentiel : `amzn-main-Base`
- ID de référentiel : `amzn-updates/latest`

Nom de référentiel : `amzn-updates-Base`

Sur Amazon Linux 2

- ID de référentiel : `amzn2-core/2/architecture`

  Nom de référentiel : `Amazon Linux 2 core repository`
- ID de référentiel : `amzn2extra-docker/2/architecture`

  Nom de référentiel : `Amazon Extras repo for docker`

> **Note**
>
> *`l'architecture`* peut être x86_64 ou aarch64.

Les instances Amazon Linux 2023 (AL2023) contiennent initialement les mises à jour disponibles dans la version d'AL2023 et l'AMI sélectionnée. Par défaut, votre instance AL2023 ne reçoit pas automatiquement de mises à jour de sécurité critiques et importantes supplémentaires au lancement. Au lieu de cela, grâce à la fonctionnalité de mises à niveau déterministes via des référentiels versionnés d'AL2023, qui est activée par défaut, vous pouvez appliquer des mises à jour selon un calendrier qui répond à vos besoins spécifiques. Pour plus d'informations, veuillez consulter la rubrique Mises à niveau déterministes via des référentiels versionnés dans le Guide de l'utilisateur Amazon Linux 2023.

Sur Amazon Linux 2022, les référentiels préconfigurés sont liés à des versions verrouillées des mises à jour des packages. Lorsque de nouvelles Amazon Machine Images (AMIs) pour Amazon Linux 2022 sont publiées, elles sont verrouillées pour une version spécifique. Pour les mises à jour des correctifs, Patch Manager récupère la dernière version verrouillée du référentiel de mises à jour des correctifs, puis met à jour les packages sur le nœud géré en fonction du contenu de cette version verrouillée.

Sur AL2023, le référentiel préconfiguré est le suivant :

- ID de référentiel : `amazonlinux`

  Nom du référentiel : référentiel Amazon Linux 2023

Sur Amazon Linux 2022 (version préliminaire), les référentiels préconfigurés sont liés à des versions verrouillées des mises à jour des packages. Lorsque de nouvelles Amazon Machine Images (AMIs) pour Amazon Linux 2022 sont publiées, elles sont verrouillées pour une version spécifique. Pour les mises à jour des correctifs, Patch Manager récupère la dernière version verrouillée du référentiel de mises à jour des correctifs, puis met à jour les packages sur le nœud géré en fonction du contenu de cette version verrouillée.

Sur Amazon Linux 2022, le référentiel préconfiguré est le suivant :

- ID de référentiel : `amazonlinux`

  Nom du référentiel : référentiel Amazon Linux 2022

> **Note**
>
> Toutes les mises à jour sont téléchargées à partir des référentiels distants configurés sur le nœud géré. Par conséquent, le nœud doit disposer d'un accès sortant à l'internet afin de se connecter aux référentiels pour que le correctif puisse être exécuté.

Les nœuds gérés Amazon Linux 1 et Amazon Linux 2 utilisent Yum comme gestionnaire de packages. Amazon Linux 2022 et Amazon Linux 2023 utilisent DNF comme gestionnaire de packages.

Les deux gestionnaires de packages utilisent le concept d'un avis de mise à jour comme 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 correctifs de sécurité par Patch Manager. Les packages ne se voient pas attribuer des classifications ou des niveaux de sévérité. Pour cette raison, Patch Manager affecte les attributs d'une notice de mise à jour aux packages associés.

> **Note**
>
> Si vous cochez la case Inclusion de mises à jour non liées à la sécurité sur la page Créer une référence de correctif, les packages qui ne sont pas classifiées dans un fichier `updateinfo.xml` (ou un package contenant un fichier sans valeurs Classification, Gravité 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 and CentOS Stream

Sur CentOS et CentOS Stream, le service de référentiel de correctifs Systems Manager utilise des référentiels préconfigurés (repos) sur le nœud géré. La liste suivante fournit des exemples pour une Amazon Machine Image (AMI) CentOS 8.2 fictive :

- ID de référentiel : `example-centos-8.2-base`

  Nom de référentiel : `Example CentOS-8.2 - Base`
- ID de référentiel : `example-centos-8.2-extras`

  Nom de référentiel : `Example CentOS-8.2 - Extras`
- ID de référentiel : `example-centos-8.2-updates`

  Nom de référentiel : `Example CentOS-8.2 - Updates`
- ID de référentiel : `example-centos-8.x-examplerepo`

  Nom de référentiel : `Example CentOS-8.x – Example Repo Packages`

> **Note**
>
> Toutes les mises à jour sont téléchargées à partir des référentiels distants configurés sur le nœud géré. Par conséquent, le nœud doit disposer d'un accès sortant à l'internet afin de se connecter aux référentiels pour que le correctif puisse être exécuté.

Les nœuds gérés CentOS 6 et 7 utilisent Yum comme gestionnaire de package. CentOS 8 et CentOS Stream l'utilisation de DNF comme gestionnaire de package par les nœuds. Les deux gestionnaires de packages utilisent le concept d'un avis de mise à jour. Une notice de mise à jour est simplement un ensemble de packages qui corrigent des problèmes spécifiques.

Toutefois, CentOS et CentOS Stream les repos par défaut ne sont pas configurés avec un avis de mise à jour. Cela signifie que Patch Manager ne détecte pas les packages sur CentOS par défaut et CentOS Stream repos. Pour permettre au Patch Manager de traiter des packages

qui ne sont pas contenus dans une notice de mise à jour, vous devez activer l'indicateur `EnableNonSecurity` dans les règles de référentiel de correctifs.

> **Note**
>
> CentOS et CentOS Stream les avis de mises à jour sont pris en charge. Les référentiels avec des notices de mise à jour peuvent être téléchargés après le lancement.

Debian Server and Raspberry Pi OS

Sous Debian Server et Raspberry Pi OS (anciennement Raspbian), le service de référentiel de correctifs Systems Manager utilise des référentiels 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 du référentiel `debian-security` *`codename`*. Cela signifie que sur chaque version deDebian Server, identifie Patch Manager uniquement les mises à niveau qui font partie du dépôt associé à cette version, comme suit :

- Debian Server 8 : `debian-security jessie`
- Debian Server9 : `debian-security stretch`
- Debian Server10 : `debian-security buster`
- Debian Server11 : `debian-security bullseye`
- Debian Server12 : `debian-security bookworm`

> **Note**
>
> Sous Debian Server 8 uniquement : dans la mesure où certains nœuds gérés de Debian Server 8.* font référence à un référentiel de packages obsolète (`jessie-backports`), Patch Manager applique des étapes supplémentaires pour s'assurer du succès des opérations d'application de correctifs. Pour plus d'informations, consultez Installation des correctifs.

Oracle Linux

Sous Oracle Linux, le service de référentiel de correctifs Systems Manager utilise des référentiels préconfigurés sur le nœud géré. Un nœud comporte généralement deux référentiels préconfigurés :

Oracle Linux 7 :

- ID de référentiel : `ol7_UEKR5/x86_64`

  Nom de référentiel : `Latest Unbreakable Enterprise Kernel Release 5 for Oracle Linux 7Server (x86_64)`
- ID de référentiel : `ol7_latest/x86_64`

  Nom de référentiel : `Oracle Linux 7Server Latest (x86_64)`

Oracle Linux 8 :

- ID de référentiel : `ol8_baseos_latest`

  Nom de référentiel : `Oracle Linux 8 BaseOS Latest (x86_64)`
- ID de référentiel : `ol8_appstream`

  Nom de référentiel : `Oracle Linux 8 Application Stream (x86_64)`
- ID de référentiel : `ol8_UEKR6`

  Nom de référentiel : `Latest Unbreakable Enterprise Kernel Release 6 for Oracle Linux 8 (x86_64)`

Oracle Linux 9 :

- ID de référentiel : `ol9_baseos_latest`

  Nom de référentiel : `Oracle Linux 9 BaseOS Latest (x86_64)`
- ID de référentiel : `ol9_appstream`

  Nom de référentiel : `Oracle Linux 9 Application Stream Packages(x86_64)`
- ID de référentiel : `ol9_UEKR7`

Nom de référentiel : `Oracle Linux UEK Release 7 (x86_64)`

> **Note**
>
> Toutes les mises à jour sont téléchargées à partir des référentiels distants configurés sur le nœud géré. Par conséquent, le nœud doit disposer d'un accès sortant à l'internet afin de se connecter aux référentiels pour que le correctif puisse être exécuté.

Les nœuds gérés Oracle Linux utilisent Yum comme gestionnaire de package, tandis que Yum utilise le concept d'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. Les packages ne se voient pas attribuer des classifications ou des niveaux de sévérité. C'est la raison pour laquelle Patch Manager affecte les attributs d'une notice de mise à jour aux packages associés et installe les packages en fonction des filtres de classification spécifiés dans le référentiel de correctif.

> **Note**
>
> Si vous cochez la case Inclusion de mises à jour non liées à la sécurité sur la page Créer une référence de correctif, les packages qui ne sont pas classifiées dans un fichier `updateinfo.xml` (ou un package contenant un fichier sans valeurs Classification, Gravité 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.

AlmaLinux, RHEL, and Rocky Linux

Activé AlmaLinuxRed Hat Enterprise Linux, et Rocky Linux le service de base de correctifs de Systems Manager utilise des référentiels préconfigurés (repos) sur le nœud géré. Un nœud comporte généralement trois référentiels préconfigurés :

Toutes les mises à jour sont téléchargées à partir des référentiels distants configurés sur le nœud géré. Par conséquent, le nœud doit disposer d'un accès sortant à l'internet afin de se connecter aux référentiels pour que le correctif puisse être exécuté.

> **Note**
>
> Si vous cochez la case Inclusion de mises à jour non liées à la sécurité sur la page Créer une référence de correctif, les packages qui ne sont pas classifiées dans un fichier `updateinfo.xml` (ou un package contenant un fichier sans valeurs Classification, Gravité 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 nœuds gérés utilisent Yum comme gestionnaire de packages. AlmaLinux, Red Hat Enterprise Linux 8, et les nœuds Rocky Linux gérés utilisent DNF comme gestionnaire de packages. Les deux gestionnaires de packages utilisent le concept d'un avis de mise à jour comme fichier nommé `updateinfo.xml`. Une notice de mise à jour est simplement un ensemble de packages qui corrigent des problèmes spécifiques. Les packages ne se voient pas attribuer des classifications ou des niveaux de sévérité. C'est la raison pour laquelle Patch Manager affecte les attributs d'une notice de mise à jour aux packages associés et installe les packages en fonction des filtres de classification spécifiés dans le référentiel de correctif.

RHEL 7

> **Note**
>
> Les ID de référentiels suivants sont associés à RHUI 2. RHUI 3 a été lancé en décembre 2019 et a introduit un schéma de d'attribution de noms différent pour les ID de référentiel Yum. En fonction de l'AMI RHEL-7 à partir de laquelle vous créez vos nœuds gérés, une mise à jour de vos commandes peut être nécessaire. Pour plus d'informations, consultez la section Repository IDs for RHEL 7 dans AWS Have Changed sur le portail client Red Hat.

- ID de référentiel : `rhui-REGION-client-config-server-7/x86_64`

  Nom de référentiel : `Red Hat Update Infrastructure 2.0 Client Configuration Server 7`
- ID de référentiel : `rhui-REGION-rhel-server-releases/7Server/x86_64`

  Nom de référentiel : `Red Hat Enterprise Linux Server 7 (RPMs)`

- ID de référentiel : `rhui-REGION-rhel-server-rh-common/7Server/x86_64`

  Nom de référentiel : `Red Hat Enterprise Linux Server 7 RH Common (RPMs)`

AlmaLinux, 8, RHEL 8 et Rocky Linux 8

- ID de référentiel : `rhel-8-appstream-rhui-rpms`

  Nom de référentiel : `Red Hat Enterprise Linux 8 for x86_64 - AppStream from RHUI (RPMs)`

- ID de référentiel : `rhel-8-baseos-rhui-rpms`

  Nom de référentiel : `Red Hat Enterprise Linux 8 for x86_64 - BaseOS from RHUI (RPMs)`

- ID de référentiel : `rhui-client-config-server-8`

  Nom de référentiel : `Red Hat Update Infrastructure 3 Client Configuration Server 8`

AlmaLinux 9, RHEL 9 et Rocky Linux 9

- ID de référentiel : `rhel-9-appstream-rhui-rpms`

  Nom de référentiel : `Red Hat Enterprise Linux 9 for x86_64 - AppStream from RHUI (RPMs)`

- ID de référentiel : `rhel-9-baseos-rhui-rpms`

  Nom de référentiel : `Red Hat Enterprise Linux 9 for x86_64 - BaseOS from RHUI (RPMs)`

- ID de référentiel : `rhui-client-config-server-9`

  Nom de référentiel : `Red Hat Enterprise Linux 9 Client Configuration`

SLES

Sur les nœuds gérés 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 nœuds gérés SLES utilisent Zypper comme gestionnaire de package, tandis que Zypper utilise le concept de correctif. Un correctif est tout simplement un ensemble de packages qui corrigent un problème spécifique. Patch Manager gère tous les packages référencés dans un correctif comme étant liés à la sécurité. Étant donné que les différents packages ne sont associés à aucune classification ou sévérité, Patch Manager leur affecte les attributs du correctif auquel ils appartiennent.

Ubuntu Server

Sous Ubuntu Server, le service de référentiel de correctifs Systems Manager utilise des référentiels préconfigurés sur le nœud géré. 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 de référentiels *`codename`*`-security`, le nom de code étant unique à la version, `trusty` pour Ubuntu Server 14 par exemple. Patch Manager identifie uniquement les mises à niveau qui font partie de ces référentiels :

- Ubuntu Server 14.04 LTS : `trusty-security`
- Ubuntu Server 16.04 LTS : `xenial-security`
- Ubuntu Server 18.04 LTS : `bionic-security`
- Ubuntu Server 20.04 LTS : `focal-security`
- Ubuntu Server 20.10 STR : `groovy-security`
- Ubuntu Server 22.04 LTS (`jammy-security`)
- Ubuntu Server23,04 () `lunar-security`

Windows Server

Sur les systèmes d'exploitation Microsoft Windows, Patch Manager récupère une liste des mises à jour disponibles que Microsoft publie dans Microsoft Update et sont automatiquement disponibles pour Windows Server Update Services (WSUS).

Patch Manager surveille 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 provenant de Microsoft sont traitées, Patch Manager supprime de la liste des correctifs les mises à jour qui ont été remplacés 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 KB4012214 est disponible en tant que mise à jour dans Patch Manager.

Patch Manager ne met à disposition que les correctifs destinés aux versions du système d'exploitation Windows Server prises en charge par Patch Manager. Par exemple, Patch Manager ne peut pas être utilisé pour appliquer un correctif à Windows RT.

> **Note**
>
> Dans certains cas, Microsoft publie des correctifs pour les applications qui ne spécifient pas de date et d'heure de mise à jour. La date et l'heure 01/01/1970 sont alors fournies par défaut.

Spécification d'un autre référentiel source de correctifs (Linux)

Lorsque vous utilisez les référentiels par défaut configurés sur un nœud géré pour les opérations d'application de correctifs, une fonctionnalité de rechercheAWS Systems Managerinstallation de correctifs liés à la sécurité. Il s'agit du comportement par défaut pour Patch Manager. Pour des informations détaillées sur la façon dont Patch Manager sélectionne et installe les correctifs de sécurité, consultez Sélection des correctifs de sécurité.

Toutefois, sur les systèmes Linux, vous pouvez également utiliser Patch Manager pour installer des correctifs non liés à la sécurité ou provenant d'un référentiel source autre de celui configuré par défaut sur le nœud géré. Vous pouvez spécifier d'autres référentiels source de correctifs lors de la création d'un référentiel de correctifs personnalisée. Dans chaque référentiel 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 gérées Ubuntu Server 14.04 et Ubuntu Server 16.04. Dans ce cas, vous pouvez spécifier d'autres référentiels pour chaque version dans la même référentiel 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'un référentiel de correctifs personnalisé spécifiant des référentiels alternatifs pour un nœud géré ne fait pas de ceux-ci les nouveaux référentiels par défaut sur le système d'exploitation. Une fois les correctifs appliqués, les référentiels précédemment configurés par défaut pour le système d'exploitation du nœud restent les référentiels par défaut.

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 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.

Exemple : utilisation de la console

Pour spécifier d'autres référentiels source de correctifs lorsque vous travaillez dans la console Systems Manager, utilisez la section Sources des correctifs de la page Créer un référentiel de correctifs. Pour de plus amples informations sur l'utilisation des options Sources des correctifs, veuillez consulter Création d'un référentiel de correctifs personnalisé (Linux).

Exemple : utilisation du AWS CLI

Pour un exemple d'utilisation de l'option `--sources` à l'aide de l' AWS Command Line Interface (AWS CLI), consultez Création d'un référentiel de correctifs avec des référentiels personnalisés pour les différentes versions du système d'exploitation.

Rubriques

- Points importants à prendre en compte pour d'autres référentiels
- Exemples d'utilisations pour d'autres référentiels source de correctifs

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 un nœud géré.

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 les nœuds gérés Amazon Linux 2, les référentiels par défaut sont `amzn2-core` et `amzn2extra-docker`. Si vous souhaitez inclure le référentiel EPEL (Extra Packages for Enterprise Linux) dans vos opérations d'application de correctifs, vous devez spécifier ces trois référentiels en tant que référentiels alternatifs.

> **Note**
>
> L'exécution d'un référentiel de correctifs personnalisé spécifiant des référentiels alternatifs pour un nœud géré ne fait pas de ceux-ci les nouveaux référentiels par défaut sur le système d'exploitation. Une fois les correctifs appliqués, les référentiels précédemment configurés par défaut pour le système d'exploitation du nœud restent les référentiels par défaut.

Le comportement d'application de correctifs pour les distributions basées sur YUM dépend du manifeste updateinfo.xml

Lorsque vous spécifiez des référentiels de correctifs alternatifs pour les distributions basées sur Yum, telles qu'Amazon Linux 1, Amazon Linux 2 Red Hat Enterprise Linux ou CentOS, le comportement des correctifs dépend du fait que le référentiel inclut ou non un manifeste de mise à jour sous la forme d'un fichier complet et correctement formaté. `updateinfo.xml` Ce fichier spécifie la date de parution, la classification et la sévérité des différents packages. Les éléments suivants ont un impact sur le comportement d'application des correctifs :

- Si vous filtrez selon Classification et Severity (Sévérité), 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 ApprovalAfterDays, mais que la date de sortie du package n'est pas au format Unix Epoch (ou qu'aucune date de sortie n'est spécifiée), le package ne sera pas inclus dans le filtre.
- Il existe une exception si vous cochez la case Inclusion de mises à jour non liées à la sécurité sur la page Créer une référence de correctif. Dans ce cas, les packages sans fichier `updateinfo.xml` (ou contenant ce fichier sans que les valeurs de Classification, Severity (Sévérité) et Date soient correctement formatées) seront inclus dans la liste préfiltrée des correctifs. (Ils doivent encore satisfaire aux autres conditions préalables relatives aux règles de référentiel 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 avez déjà l'habitude d'Patch Managerinstaller des correctifs de sécurité sur un parc de nœuds Ubuntu Server gérés à l'aide de la base `AWS-UbuntuDefaultPatchBaseline` de correctifs prédéfinie AWS fournie. Vous pouvez créer un référentiel 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 ce référentiel de correctifs est exécuté sur vos nœuds, tous les correctifs, qu'ils soient liés ou non à la sécurité, sont 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 nœuds gérés Ubuntu Server exécutent des logiciels distribués par le biais de référentiels PPA (Personal Package Archive) pour Ubuntu. Dans ce cas, vous créez un référentiel de correctifs spécifiant un référentiel PPA que vous avez configuré sur le nœud géré en tant que référentiel source pour l'application des correctifs. Utilisez ensuite la fonctionnalité Run Command pour exécuter le document de référentiel de correctifs sur les nœuds.

Exemple 3 – Applications d'entreprise internes sur Amazon Linux

Certaines applications doivent être exécutées sur vos nœuds gérés Amazon Linux pour mettre ceux-ci en conformité avec les réglementations du secteur. Vous pouvez configurer un référentiel dédié à ces applications sur les nœuds, utiliser YUM pour installer les applications, puis mettre à jour ou créer un référentiel de correctifs pour inclure ce nouveau référentiel d'entreprise. Vous pouvez ensuite utiliser la fonctionnalité Run Command pour exécuter le document `AWS-RunPatchBaseline` avec l'option `Scan` afin de déterminer si le package d'entreprise est répertorié parmi les packages installés et à jour sur le nœud géré. 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

Patch Manager, une fonctionnalité de AWS Systems Manager, utilise le mécanisme intégré approprié à un type de système d'exploitation pour installer les mises à jour sur un nœud géré. Par exemple, surWindows Server, l'API Windows Update est utilisée, et sur Amazon Linux 2, le gestionnaire de `yum` packages est utilisé.

Le reste de cette section explique comment Patch Manager installe des correctifs sur un système d'exploitation.

Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022, and Amazon Linux 2023

Sur les nœuds gérés Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022 et Amazon Linux 2023, le flux de travail d'installation des correctifs est le suivant :

1. Si une liste de correctifs est spécifiée à l'aide d'une URL https ou d'une URL de type chemin Amazon Simple Storage Service (Amazon S3) en utilisant le paramètre `InstallOverrideList` pour les documents `AWS-RunPatchBaseline` ou `AWS-RunPatchBaselineAssociation`, les correctifs répertoriés sont installés et les étapes 2 à 7 sont ignorées.
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.

   Cependant, les règles d'approbation sont également assujetties au fait que la case Inclure les mises à jour non liées à la sécurité a été cochée ou non lors de la création ou de la dernière mise à jour d'un référentiel 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 des mises à jour non liées à la sécurité sont incluses, les correctifs provenant d'autres référentiels sont également pris en compte.
4. Appliquez ApprovedPatches 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. Si plusieurs versions d'un correctif sont approuvées, la version la plus récente sera appliquée.
7. L'API de mise à jour YUM (Amazon Linux 1, Amazon Linux 2) ou l'API de mise à jour DNF (Amazon Linux 2022, Amazon Linux 2023) est appliquée aux correctifs approuvés comme suit :

- Pour les référentiels de correctifs par défaut prédéfinis fournis par AWS, seuls les correctifs spécifiés dans `updateinfo.xml` sont appliqués (mises à jour de sécurité uniquement). Cela est dû au fait que la case Inclusion de mises à jour non liées à la sécurité n'est pas cochée. Les références prédéfinies sont équivalentes à une référence personnalisée avec les éléments suivants :
  - La case Inclusion de mises à jour non liées à la sécurité n'est pas cochée.
  - Une liste de GRAVITÉ `[Critical, Important]`
  - Une liste de CLASSIFICATION `[Security, Bugfix]`

  Pour Amazon Linux 1 et Amazon Linux 2, la commande yum équivalente pour ce flux de travail est la suivante :

  ```
  sudo yum update-minimal --sec-severity=critical,important --bugfix -y
  ```

  Pour Amazon Linux 2022 et Amazon Linux 2023, la commande dnf équivalente pour ce flux de travail est :

  ```
  sudo dnf upgrade-minimal --sec-severity=critical --sec-severity=important --
  bugfix -y
  ```

  Si la case Inclusion de mises à jour non liées à la sécurité est cochée, 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 Amazon Linux 1 et Amazon Linux 2, si une ligne de base avec Inclure les mises à jour non liées à la sécurité est sélectionnée, comporte une liste de gravité `[Critical, Important]` et une liste de CLASSIFICATION de`[Security, Bugfix]`, la commande yum équivalente est la suivante :

  ```
  sudo yum update --security --sec-severity=critical,important --bugfix -y
  ```

  Pour Amazon Linux 2022 et Amazon Linux 2023, la commande dnf équivalente est :

  ```
  sudo dnf upgrade --security --sec-severity=critical --sec-severity=important --
  bugfix -y
  ```

> **Note**
>
> Pour Amazon Linux 2022 et Amazon Linux 2023, un niveau de sévérité des correctifs `Medium` équivaut à un niveau de sévérité `Moderate` pouvant être défini dans certains référentiels externes. Si vous incluez des correctifs de gravité `Medium` dans le référentiel de correctifs, des correctifs de gravité `Moderate` provenant de correctifs externes sont également installés sur les instances.
> Lorsque vous recherchez des données de conformité à l'aide de l'action d'API DescribeInstancePatches, le filtrage selon le niveau de gravité `Medium` signale les correctifs dont les niveaux de gravité sont à la fois `Medium` et `Moderate`.
> Amazon Linux 2022 et Amazon Linux 2023 prennent également en charge le niveau de sévérité des correctifs None, qui est reconnu par le gestionnaire de packages DNF.

8. Si des mises à jour ont été installées, le nœud géré est redémarré. (Exception : si le paramètre `RebootOption` est défini sur `NoReboot` dans le document `AWS-RunPatchBaseline`, le nœud géré n'est pas redémarré après l'exécution de Patch Manager. Pour plus d'informations, consultez Nom du paramètre: `RebootOption`).

CentOS and CentOS Stream

Sur CentOS et CentOS Stream les noeuds gérés, le flux d'installation des correctifs est le suivant :

1. Si une liste de correctifs est spécifiée à l'aide d'une URL https ou d'une URL de type chemin Amazon Simple Storage Service (Amazon S3) en utilisant le paramètre `InstallOverrideList` pour les documents `AWS-RunPatchBaseline` ou `AWS-RunPatchBaselineAssociation`, les correctifs répertoriés sont installés et les étapes 2 à 7 sont ignorées.

   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.

Cependant, les règles d'approbation sont également assujetties au fait que la case Inclure les mises à jour non liées à la sécurité a été cochée ou non lors de la création ou de la dernière mise à jour d'un référentiel 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 des mises à jour non liées à la sécurité sont incluses, les correctifs provenant d'autres référentiels sont également pris en compte.

3. Appliquez ApprovedPatches 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 and 7.x) ou la mise à jour DNF (sur CentOS 8 et CentOS Stream) applicable aux correctifs approuvés.
7. Si des mises à jour ont été installées, le nœud géré est redémarré. (Exception : si le paramètre `RebootOption` est défini sur `NoReboot` dans le document `AWS-RunPatchBaseline`, le nœud géré n'est pas redémarré après l'exécution de Patch Manager. Pour plus d'informations, consultez Nom du paramètre: `RebootOption`).

Debian Server and Raspberry Pi OS

Sur les instances Debian Server et Raspberry Pi OS (anciennement Raspbian), le flux d'installation de correctifs est le suivant :

1. Si une liste de correctifs est spécifiée à l'aide d'une URL https ou d'une URL de style chemin Amazon Simple Storage Service (Amazon S3) en utilisant le paramètre `InstallOverrideList` pour les documents `AWS-RunPatchBaseline` ou `AWS-RunPatchBaselineAssociation`, les correctifs répertoriés sont installés et les étapes 2 à 7 sont ignorées.
2. Si une mise à jour est disponible pour `python3-apt` (une interface de bibliothèque Python pour `libapt`), la mise à niveau est faite à la dernière version. (Ce package non lié à la sécurité

est mis à niveau même si vous n'avez pas sélectionné l'option Inclure les mises à jour non liées à la sécurité.)

> **Important**
>
> Sous Debian Server 8 uniquement : dans la mesure où certains nœuds gérés de Debian Server 8.* font référence à un référentiel de packages obsolète (`jessie-backports`), Patch Manager applique des étapes supplémentaires suivantes pour s'assurer du succès des opérations d'application de correctifs :
>
> a. Sur votre nœud géré, la référence au référentiel `jessie-backports` est commentée à partir de la liste des emplacements sources (`/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. L'opération `apt-get` est exécutée à ce moment précis 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.
>
> Lorsque Patch Manager met à jour à nouveau le système, le même processus est répété.

3. Appliquez des filtres GlobalFilters comme indiqué dans la référence de correctif en conservant uniquement les packages qualifiés à des fins de traitement ultérieur.
4. Appliquez ApprovalRules comme indiqué dans la référence de correctif. Chaque règle d'approbation peut définir un package comme approuvé.

> **Note**
>
> Dans la mesure où 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.

Cependant, les règles d'approbation sont également assujetties au fait que la case Inclure les mises à jour non liées à la sécurité a été cochée ou non lors de la création ou de la dernière mise à jour d'un référentiel 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 des 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 et Raspberry Pi OS, les versions de correctifs candidates se limitent aux correctifs inclus dans `debian-security`.

5. Appliquez ApprovedPatches 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.
6. 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.
7. La bibliothèque APT permet de mettre à niveau les packages.

> **Note**
>
> Patch Managerne prend pas en charge l'utilisation de l'`Pin-Priority`option APT pour attribuer des priorités aux paquets. Patch Managerregroupe les mises à jour disponibles depuis tous les référentiels activés et sélectionne la mise à jour la plus récente correspondant à la ligne de base de chaque package installé.

8. Si des mises à jour ont été installées, le nœud géré est redémarré. (Exception : si le paramètre `RebootOption` est défini sur `NoReboot` dans le document `AWS-RunPatchBaseline`, le nœud géré n'est pas redémarré après l'exécution de Patch Manager. Pour plus d'informations, consultez Nom du paramètre: `RebootOption`).

macOS

Sur les nœuds gérés macOS, le flux d'installation des correctifs est le suivant :

1. La liste de propriétés `/Library/Receipts/InstallHistory.plist` est un registre des logiciels qui ont été installés et mis à niveau à l'aide des gestionnaires de package `softwareupdate` et `installer`. La liste est analysée avec l'outil de ligne de commande `pkgutil` (pour`installer`) et les commandes CLI du gestionnaire de packages `softwareupdate`.

   Pour `installer`, la réponse aux commandes CLI comprend les détails `package name`, `version`, `volume`, `location` et `install-time`, mais Patch Manager utilise seulement le `package name` et la `version`.

   Pour `softwareupdate`, la réponse aux commandes CLI comprend le nom du package (`display name`), la `version` et la `date`, mais Patch Manager utilise seulement le nom du package et la version.

   Pour Brew et Brew Cask, Homebrew ne prend pas en charge les commandes qui s'exécutent sous l'utilisateur racine. Par conséquent, Patch Manager interroge et exécute les commandes Homebrew en tant que le propriétaire du répertoire Homebrew ou l'utilisateur valide appartenant au groupe de propriétaires du répertoire Homebrew. Les commandes sont similaires à `softwareupdate` et `installer`, et sont exécutées via un sous-processus Python pour collecter les données de package, la sortie étant analysée pour identifier les noms et les versions des packages.
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.
4. Appliquez ApprovedPatches 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. Si plusieurs versions d'un correctif sont approuvées, la version la plus récente sera appliquée.
7. Appelle la CLI du package sur le nœud géré pour traiter les correctifs approuvés comme suit :

> **Note**
>
> `installer` ne dispose pas de la fonctionnalité nécessaire pour rechercher et installer des mises à jour. Par conséquent, pour `installer`, Patch Manager signale uniquement les packages qui sont installés. Il s'ensuit que les packages `installer` ne sont jamais signalés comme `Missing`.

- Pour les référentiels de correctifs par défaut prédéfinis fournis par AWS et pour les référentiels de correctifs personnalisés pour lesquels la case Inclusion de mises à jour non liées à la sécurité n'est pas cochée, seules les mises à jour de sécurité sont appliquées.
- Pour les référentiels de correctifs personnalisés pour lesquels la case Inclusion de mises à jour non liées à la sécurité est cochée, les mises à jour de sécurité et non liées à la sécurité sont appliquées.

8. Si des mises à jour ont été installées, le nœud géré est redémarré. (Exception : si le paramètre `RebootOption` est défini sur `NoReboot` dans le document `AWS-RunPatchBaseline`, le nœud géré n'est pas redémarré après l'exécution de Patch Manager. Pour plus d'informations, consultez Nom du paramètre: `RebootOption`).

Oracle Linux

Sur les nœuds gérés Oracle Linux, le flux d'installation des correctifs est le suivant :

1. Si une liste de correctifs est spécifiée à l'aide d'une URL https ou d'une URL de type chemin Amazon Simple Storage Service (Amazon S3) en utilisant le paramètre `InstallOverrideList` pour les documents `AWS-RunPatchBaseline` ou `AWS-RunPatchBaselineAssociation`, les correctifs répertoriés sont installés et les étapes 2 à 7 sont ignorées.
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.

   Cependant, les règles d'approbation sont également assujetties au fait que la case Inclure les mises à jour non liées à la sécurité a été cochée ou non lors de la création ou de la dernière mise à jour d'un référentiel 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 des mises à jour non liées à la sécurité sont incluses, les correctifs provenant d'autres référentiels sont également pris en compte.

4. Appliquez ApprovedPatches 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. Si plusieurs versions d'un correctif sont approuvées, la version la plus récente sera appliquée.
7. Sur les nœuds gérés de version 7, l'API de mise à jour YUM est appliquée aux correctifs approuvés comme suit :

   - Pour les référentiels de correctifs par défaut prédéfinis fournis par AWS et pour les référentiels de correctifs personnalisés pour lesquels la case Inclusion de 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 référentiels de correctifs personnalisés pour lesquels la case Inclusion de mises à jour non liées à la sécurité est cochée, 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
     ```

     Sur les nœuds gérés de version 8 et 9, l'API de mise à jour DNF est appliquée aux correctifs approuvés comme suit :

- Pour les lignes de base de correctifs par défaut prédéfinies fournies par AWS, et pour les lignes de base de correctifs personnalisées pour lesquelles la case Inclure les 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 dnf upgrade-minimal --security --sec-severity=Moderate --sec-
  severity=Important
  ```

- Pour les référentiels de correctifs personnalisés pour lesquels la case Inclusion de mises à jour non liées à la sécurité est cochée, 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 dnf upgrade --security --bugfix
  ```

8. Si des mises à jour ont été installées, le nœud géré est redémarré. (Exception : si le paramètre `RebootOption` est défini sur `NoReboot` dans le document `AWS-RunPatchBaseline`, le nœud géré n'est pas redémarré après l'exécution de Patch Manager. Pour plus d'informations, consultez Nom du paramètre: `RebootOption`).

AlmaLinux, RHEL, and Rocky Linux

Sur AlmaLinux et sur les nœuds Rocky Linux gérés, le flux de travail d'installation des correctifs est le suivant : Red Hat Enterprise Linux

1. Si une liste de correctifs est spécifiée à l'aide d'une URL https ou d'une URL de type chemin Amazon Simple Storage Service (Amazon S3) en utilisant le paramètre `InstallOverrideList` pour les documents `AWS-RunPatchBaseline` ou `AWS-RunPatchBaselineAssociation`, les correctifs répertoriés sont installés et les étapes 2 à 7 sont ignorées.
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.

Cependant, les règles d'approbation sont également assujetties au fait que la case Inclure les mises à jour non liées à la sécurité a été cochée ou non lors de la création ou de la dernière mise à jour d'un référentiel 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 des mises à jour non liées à la sécurité sont incluses, les correctifs provenant d'autres référentiels sont également pris en compte.

4. Appliquez ApprovedPatches 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. Si plusieurs versions d'un correctif sont approuvées, la version la plus récente sera appliquée.
7. L'API de mise à jour YUM (sur RHEL 7) ou l'API de mise à jour DNF (sur AlmaLinux 8 et 9, RHEL 8 et 9, et Rocky Linux 8 et 9) est appliquée aux correctifs approuvés comme suit :
   - Pour les référentiels de correctifs par défaut prédéfinis fournis par AWS et pour les référentiels de correctifs personnalisés pour lesquels la case Inclusion de 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 AlmaLinux, RHEL 8 etRocky Linux, 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 référentiels de correctifs personnalisés pour lesquels la case Inclusion de mises à jour non liées à la sécurité est cochée, 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 AlmaLinux 8 et 9, RHEL 8 et 9, et Rocky Linux 8 et 9, la commande dnf équivalente pour ce flux de travail est la suivante :

```
sudo dnf update --security --bugfix -y
```

8. Si des mises à jour ont été installées, le nœud géré est redémarré. (Exception : si le paramètre `RebootOption` est défini sur `NoReboot` dans le document `AWS-RunPatchBaseline`, le nœud géré n'est pas redémarré après l'exécution de Patch Manager. Pour plus d'informations, consultez Nom du paramètre: `RebootOption`).

SLES

Sur les nœuds gérés SUSE Linux Enterprise Server (SLES), le flux d'installation des correctifs est le suivant :

1. Si une liste de correctifs est spécifiée à l'aide d'une URL https ou d'une URL de type chemin Amazon Simple Storage Service (Amazon S3) en utilisant le paramètre `InstallOverrideList` pour les documents `AWS-RunPatchBaseline` ou `AWS-RunPatchBaselineAssociation`, les correctifs répertoriés sont installés et les étapes 2 à 7 sont ignorées.
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.

   Cependant, les règles d'approbation sont également assujetties au fait que la case Inclure les mises à jour non liées à la sécurité a été cochée ou non lors de la création ou de la dernière mise à jour d'un référentiel 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 des mises à jour non liées à la sécurité sont incluses, les correctifs provenant d'autres référentiels sont également pris en compte.

4. Appliquez ApprovedPatches 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. Si plusieurs versions d'un correctif sont approuvées, la version la plus récente sera appliquée.
7. L'API de mise à jour Zypper est appliquée aux correctifs approuvés.
8. Si des mises à jour ont été installées, le nœud géré est redémarré. (Exception : si le paramètre `RebootOption` est défini sur `NoReboot` dans le document `AWS-RunPatchBaseline`, le nœud géré n'est pas redémarré après l'exécution de Patch Manager. Pour plus d'informations, consultez Nom du paramètre: `RebootOption`).

Ubuntu Server

Sur les nœuds gérés Ubuntu Server, le flux d'installation des correctifs est le suivant :

1. Si une liste de correctifs est spécifiée à l'aide d'une URL https ou d'une URL de style chemin Amazon Simple Storage Service (Amazon S3) en utilisant le paramètre `InstallOverrideList` pour les documents `AWS-RunPatchBaseline` ou `AWS-RunPatchBaselineAssociation`, les correctifs répertoriés sont installés et les étapes 2 à 7 sont ignorées.
2. Si une mise à jour est disponible pour `python3-apt` (une interface de bibliothèque Python pour `libapt`), la mise à niveau est faite à la dernière version. (Ce package non lié à la sécurité est mis à niveau même si vous n'avez pas sélectionné l'option Inclure les mises à jour non liées à la sécurité.)
3. Appliquez des filtres GlobalFilters comme indiqué dans la référence de correctif en conservant uniquement les packages qualifiés à des fins de traitement ultérieur.
4. 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.

Cependant, les règles d'approbation sont également assujetties au fait que la case Inclure les mises à jour non liées à la sécurité a été cochée ou non lors de la création ou de la dernière mise à jour d'un référentiel 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 des mises à jour non liées à la sécurité sont incluses, les correctifs provenant d'autres référentiels sont également pris en compte.

Cependant, les règles d'approbation sont également assujetties au fait que la case Inclure les mises à jour non liées à la sécurité a été cochée ou non lors de la création ou de la dernière mise à jour d'un référentiel de correctif.

> **Note**
>
> Pour chaque version deUbuntu Server, les versions candidates aux correctifs sont limitées aux correctifs qui font partie du dépôt associé à cette version, comme suit :
>
> - Ubuntu Server 14.04 LTS : `trusty-security`
> - Ubuntu Server 16.04 LTS : `xenial-security`
> - Ubuntu Server 18.04 LTS : `bionic-security`
> - Ubuntu Server20,04 LITRES) : `focal-security`
> - Ubuntu Server 20.10 STR : `groovy-security`
> - Ubuntu Server22,04 LTS : `jammy-security`
> - Ubuntu Server23,04 : `lunar-lobster`

5. Appliquez ApprovedPatches 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.
6. 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.
7. La bibliothèque APT permet de mettre à niveau les packages.

   > **Note**
   >
   > Patch Managerne prend pas en charge l'utilisation de l'`Pin-Priority`option APT pour attribuer des priorités aux paquets. Patch Managerregroupe les mises à jour disponibles depuis tous les référentiels activés et sélectionne la mise à jour la plus récente correspondant à la ligne de base de chaque package installé.

8. Si des mises à jour ont été installées, le nœud géré est redémarré. (Exception : si le paramètre `RebootOption` est défini sur `NoReboot` dans le document `AWS-RunPatchBaseline`, le nœud géré n'est pas redémarré après l'exécution de Patch Manager. Pour plus d'informations, consultez Nom du paramètre: `RebootOption`).

Windows Server

Lors de l'application de correctifs sur un nœud géré Windows Server, celui-ci demande à Systems Manager un instantané du référentiel de correctifs approprié. Cet instantané contient la liste de toutes les mises à jour disponibles dans le référentiel de correctifs ayant été approuvées à des fins de déploiement. Cette liste est envoyée à l'API Windows Update, qui détermine quelles mises à jour sont applicables au nœud géré et, si nécessaire, les installe. Si des mises à jour sont installées, le nœud géré est ensuite redémarré autant de fois que nécessaire pour appliquer les correctifs requis. (Exception : si le paramètre `RebootOption` est défini sur `NoReboot` dans le document `AWS-RunPatchBaseline`, le nœud géré n'est pas redémarré après l'exécution de Patch Manager. Pour plus d'informations, consultez Nom du paramètre: `RebootOption`). La synthèse de l'opération d'application de correctifs se situe dans la sortie de la demande Run Command. Des journaux supplémentaires sont disponibles dans le dossier `%PROGRAMDATA%\Amazon\PatchBaselineOperations\Logs` du nœud géré.

Du fait que l'API Windows Update est utilisée pour télécharger et installer des correctifs, tous les paramètres de Politique de groupe de Windows Update sont respectés. Aucun paramètre lié à la politique de groupe n'est requis pour utiliser Patch Manager, mais tous les paramètres que vous

avez définis seront appliqués, par exemple pour diriger les nœuds gérés vers le serveur WSUS (Windows Server Update Services).

> **Note**
>
> Par défaut, Windows télécharge tous les correctifs sur le site Windows Update de Microsoft, car Patch Manager utilise l'API Windows Update pour guider le téléchargement et l'installation des correctifs. Par conséquent, le nœud géré doit avoir accès au site Microsoft Windows Update, faute de quoi l'application des correctifs échouera. Vous pouvez également configurer un serveur WSUS en tant que référentiel de correctifs, et configurer vos nœuds gérés pour qu'ils ciblent ce serveur WSUS à l'aide de politiques de groupe.

Fonctionnement des règles de référence de correctif sur les systèmes basés sur Linux

Les règles d'un référentiel de correctif pour les distributions Linux fonctionnent différemment en fonction du type de distribution. Contrairement aux mises à jour de correctifs sur les nœuds Windows Server gérés, les règles sont évaluées sur chaque nœud afin de prendre en compte les dépôts configurés sur l'instance. Patch Manager, une fonctionnalité de AWS Systems Manager, utilise le gestionnaire de packages natif pour piloter l'installation des correctifs approuvés par la base de correctifs.

Pour les types de système d'exploitation basés sur Linux qui signalent un niveau de sévérité pour les correctifs, Patch Manager utilise le niveau de sévérité signalé par l'éditeur du logiciel pour l'avis de mise à jour ou le correctif individuel. Patch Manager ne dérive pas les niveaux de sévérité de sources tierces, telles que le Common Vulnerability Scoring System (CVSS), ou des métriques publiées par la National Vulnerability Database (NVD).

Rubriques

- Comment fonctionnent les règles de base des correctifs sur Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022 et Amazon Linux 2023
- Le fonctionnement des règles liées au référentiel de correctifs sur CentOS et CentOS Stream
- Fonctionnement des règles de référence de correctifs sur Debian Server et Raspberry Pi OS
- Fonctionnement des règles de référence de correctif sur macOS
- Fonctionnement des règles de référence de correctif sur Oracle Linux
- Comment fonctionnent les règles de base des correctifs sur AlmaLinuxRHEL, et Rocky Linux

- Fonctionnement des règles de référence de correctif sur SUSE Linux Enterprise Server
- Fonctionnement des règles de référence de correctif sur Ubuntu Server

Comment fonctionnent les règles de base des correctifs sur Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022 et Amazon Linux 2023

Sur Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022 et Amazon Linux 2023, le processus de sélection des correctifs est le suivant :

1. Sur le nœud géré, la bibliothèque YUM (Amazon Linux 1 et Amazon Linux 2) ou la bibliothèque DNF (Amazon Linux 2022 et Amazon Linux 2023) accède au `updateinfo.xml` fichier pour chaque dépôt configuré.

   > **Note**
   >
   > Si aucun fichier `updateinfo.xml` n'est trouvé, l'installation de correctifs dépend des paramètres Inclusion de mises à jour non liées à la sécurité et Approbation automatique. Par exemple, si les mises à jour non liées à la sécurité sont autorisées, elles sont installées lorsque l'heure de l'approbation automatique arrive.

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 consulter la liste des valeurs prises en charge à l'aide de la AWS CLI commande describe-patch-properties ou de l'opération APIDescribePatchProperties. Vous pouvez aussi afficher la liste dans la zone

Attribut	Description
	Règles d'approbation de la page Créer un référentiel de correctif ou la page Modifier le référentiel 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 consulter la liste des valeurs prises en charge à l'aide de la AWS CLI commande describe-patch-properties ou de l'opération APIDescribePatchProperties. Vous pouvez aussi afficher la liste dans la zone Règles d'approbation de la page Créer un référentiel de correctif ou la page Modifier le référentiel 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 ApprovedPatches ou RejectedP atches, 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 ApprovedP atches ou RejectedPatches, dans la référence de correctifs.

Attribut	Description
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 afin de déterminer l'approbation des correctifs à des fins de déploiement.

> **Note**
>
> Pour obtenir des informations sur les formats acceptés de 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.

3. Le produit du nœud géré est déterminé par SSM Agent. 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
Référentiels de correctifs par défaut prédéfinis fournis par AWS et référentiels de correctifs personnalisés pour lesquels l'option Inclusion de mises à jour non liées à la sécurité n'est pas sélectionnée	Pour chaque notice de mise à jour dans `updateinfo.xml` , le référentiel 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 Amazon Linux 1 et Amazon Linux 2, la commande yum équivalente pour ce flux de travail est la suivante :

Option de sécurité	Sélection de correctifs
	`sudo yum update-minimal --sec-severity=Critical,Important --bugfix -y` Pour Amazon Linux 2022 et Amazon Linux 2023, la commande dnf équivalente pour ce flux de travail est : `sudo dnf upgrade-minimal --sec-severity=Critical --sec-severity=Important --bugfix -y`
Référentiels de correctifs personnalisés pour lesquels la case Inclusion de mises à jour non liées à la sécurité est cochée avec une liste de GRAVITÉ `[Critical, Important]` et une liste de CLASSIFICATION `[Security, Bugfix]`	En plus de l'application des mises à jour de sécurité qui ont été sélectionnées à partir de `updateinfo.xml` , Patch Manager applique les mises à jour non liées à la sécurité qui remplissent, par ailleurs, les règles de filtrage des correctifs. Pour Amazon Linux et Amazon Linux 2, la commande yum équivalente pour ce flux de travail est : `sudo yum update-minimal --security --sec-severity=Critical,Important --bugfix -y` Pour Amazon Linux 2022 et Amazon Linux 2023, la commande dnf équivalente pour ce flux de travail est : `sudo dnf upgrade-minimal --security --sec-severity=Critical --sec-severity=Imporant --bugfix -y`

Pour de plus amples informations sur les valeurs du statut de conformité des correctifs, veuillez consulter Comprendre les valeurs d'état de conformité des correctifs.

Le fonctionnement des règles liées au référentiel de correctifs sur CentOS et CentOS Stream

CentOS et les référentiels CentOS Stream par défaut n'incluent aucun fichier. `updateinfo.xml` Toutefois, les référentiels personnalisés que vous créez ou utilisez peuvent inclure ce fichier. Dans cette rubrique, les références ne s'`updateinfo.xml`appliquent qu'à ces référentiels personnalisés.

Sur CentOS et CentOS Stream, le processus de sélection des correctifs est le suivant:

1. Sur le nœud géré, la bibliothèque YUM (sur les versions de CentOS 6.x et 7.x) ou la bibliothèque DNF (sur CentOS 8.x et CentOS Stream) `updateinfo.xml` accède au fichier, s'il existe dans un référentiel personnalisé, pour chaque dépôt configuré.

   Si aucun correctif n'est `updateinfo.xml` trouvé, ce qui inclut toujours les dépôts par défaut, l'installation des correctifs dépend des paramètres d'inclusion des mises à jour non liées à la sécurité et d'approbation automatique. Par exemple, si les mises à jour non liées à la sécurité sont autorisées, elles sont installées lorsque l'heure de l'approbation automatique arrive.
2. S'il `updateinfo.xml` est présent, chaque avis de mise à jour du fichier inclut plusieurs attributs qui indiquent les propriétés des packages figurant dans l'avis, 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 consulter la liste des valeurs prises en charge à l'aide de la AWS CLI commande describe-patch-properties ou de l'opération APIDescribePatchProperties. Vous pouvez aussi afficher la liste dans la zone Règles d'approbation de la page Créer un référentiel de correctif ou la page Modifier

Attribut	Description
	le référentiel 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 consulter la liste des valeurs prises en charge à l'aide de la AWS CLI commande describe-patch-properties ou de l'opération APIDescribePatchProperties. Vous pouvez aussi afficher la liste dans la zone Règles d'approbation de la page Créer un référentiel de correctif ou la page Modifier le référentiel 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 ApprovedPatches 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 ApprovedPatches ou RejectedPatches, dans la référence de correctifs.

Attribut	Description
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 afin de déterminer l'approbation des correctifs à des fins de déploiement.

> **Note**
>
> Pour obtenir des informations sur les formats acceptés de 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.

3. Dans tous les cas, le produit du nœud géré est déterminé parSSM Agent. 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
Référentiels de correctifs par défaut prédéfinis fournis par AWS et référentiels de correctifs personnalisés pour lesquels l'option Inclusion de mises à jour non liées à la sécuritén'est pas sélectionnée	Pour chaque avis de mise à jour`updateinfo.xml` , s'il existe dans un référentiel personnalisé, la ligne de base de correctif est utilisée comme filtre, permettant uniquement d'inclure les packages qualifiés 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 6 et 7 où `updateinfo.xml` est présent, la commande yum équivalente pour ce flux de travail est la suivante : `sudo yum update-minimal --sec-severity=Critical,Important --bugfix -y` Pour CentOS 8 et CentOS Stream ses emplacements, la commande dnf équivalente pour ce flux de travail `updateinfo.xml` est la suivante : `sudo dnf upgrade-minimal --sec-severity=Critical --sec-severity=Important --bugfix -y`

Option de sécurité	Sélection de correctifs
Référentiels de correctifs personnalisés pour lesquels la case Inclusion de mises à jour non liées à la sécurité est cochée avec une liste de GRAVITÉ `[Critical, Important]` et une liste de CLASSIFICATION `[Security, Bugfix]`	Outre l'application des mises à jour de sécurité sélectionnées`updateinfo.xml`, s'il existe dans un référentiel personnalisé, Patch Manager applique les mises à jour non liées à la sécurité qui respectent par ailleurs les règles de filtrage des correctifs. Pour CentOS 6 et 7 où `updateinfo.xml` est présent, la commande yum équivalente pour ce flux de travail est la suivante : `sudo yum update --sec-severity=Critical,Important --bugfix -y` Pour CentOS 8 et CentOS Stream ses emplacements, la commande dnf équivalente pour ce flux de travail `updateinfo.xml` est la suivante : `sudo dnf upgrade --security --sec-severity=Critical --sec-severity=Important --bugfix -y` Pour les dépôts par défaut et les dépôts personnalisés sans`updateinfo.xml`, vous devez cocher la case Inclure les mises à jour non liées à la sécurité afin de mettre à jour les packages du système d'exploitation (OS).

Pour de plus amples informations sur les valeurs du statut de conformité des correctifs, veuillez consulter Comprendre les valeurs d'état de conformité des correctifs.

Fonctionnement des règles de référence de correctifs sur Debian Server et Raspberry Pi OS

Sur Debian Server et Raspberry Pi OS (anciennement Raspbian), 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 et Raspberry Pi OS. Pour déterminer si un correctif est sélectionné par le référentiel de correctif, Patch Manager effectue les opérations suivantes :

1. Sous les systèmes Debian Server et Raspberry Pi OS, 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. Si une mise à jour est disponible pour `python3-apt` (une interface de bibliothèque Python pour `libapt`), la mise à niveau est faite à la dernière version. (Ce package non lié à la sécurité est mis à niveau même si vous n'avez pas sélectionné l'option Inclure les mises à jour non liées à la sécurité.)

   > **Important**
   >
   > Sur Debian Server 8 uniquement : étant donné que certains systèmes d'exploitation Debian Server 8.* font référence à un référentiel de packages obsolète (`jessie-backports`), Patch Manager effectue des étapes supplémentaires pour s'assurer que les opérations d'application de correctifs réussissent :
   >
   > a. Sur votre nœud géré, la référence au référentiel `jessie-backports` est commentée à partir de la liste des emplacements sources (`/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. L'opération `apt-get` est exécutée à ce moment précis 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.

3. Ensuite, les listes GlobalFilters, ApprovalRules, ApprovedPatches et RejectedPatches sont appliquées.

> **Note**
>
> Dans la mesure où 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.

Cependant, les règles d'approbation sont également assujetties au fait que la case Inclure les mises à jour non liées à la sécurité a été cochée ou non lors de la création ou de la dernière mise à jour d'un référentiel 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. Dans ce cas, pour Debian Server les versions de correctifs candidates se limitent aux correctifs inclus dans les référentiels suivants :

La dénomination de ces référentiels est la suivante :

- Debian Server 8 : `debian-security jessie`
- Debian Server et Raspberry Pi OS 9 : `debian-security stretch`
- Debian Server10 : `debian-security buster`
- Debian Server11 : `debian-security bullseye`
- Debian Server12 : `debian-security bookworm`

Si des mises à jour non liées à la sécurité sont incluses, les correctifs provenant d'autres référentiels sont également pris en compte.

> **Note**
>
> Pour obtenir des informations sur les formats acceptés pour les listes de correctifs approuvés et de correctifs rejetés, veuillez consulter À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés.

Pour afficher le contenu des champs Priorité et Section, exécutez la commande `aptitude` suivante :

> **Note**
>
> Il se peut que vous deviez d'abord installer Aptitude sur les systèmes Debian Server.

```
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 Comprendre les valeurs d'état de conformité des correctifs.

Fonctionnement des règles de référence de correctif sur macOS

Sur les macOS, le processus de sélection des correctifs est le suivant :

1. Sur le nœud géré, Patch Manager accède au contenu analysé du fichier `InstallHistory.plist` et identifie les noms et les versions des packages.

   Pour obtenir des détails sur le processus d'analyse, veuillez consulter la section macOS dans Installation des correctifs.
2. Le produit du nœud géré est déterminé par SSM Agent. Cet attribut correspond à la valeur de l'attribut de la clé de produit du type de données PatchFilter de la référence de correctif.
3. Les packages sont sélectionnés pour la mise à jour en suivant les consignes suivantes.

Option de sécurité	Sélection de correctifs
Référentiels de correctifs par défaut prédéfinis fournis par AWS et référentiels de correctifs personnalisés pour lesquels l'option Inclusion de mises à jour non liées à la sécuritén'est pas sélectionnée	Pour chaque mise à jour de package disponible, le référentiel de correctifs est utilisé 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

Option de sécurité	Sélection de correctifs
	de correctif, la version la plus récente est utilisée.
Référentiels de correctifs personnalisés pour lesquels l'option Inclusion de mises à jour non liées à la sécurité est sélectionnée	En plus de l'application des mises à jour de sécurité qui ont été identifiées à l'aide de `InstallHistory.plist` , Patch Manager applique les mises à jour non liées à la sécurité qui remplissent, par ailleurs, les règles de filtrage des correctifs.

Pour de plus amples informations sur les valeurs du statut de conformité des correctifs, veuillez consulter Comprendre les valeurs d'état de conformité des correctifs.

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 le nœud géré, la bibliothèque YUM 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 Oracle. Si aucun fichier `updateinfo.xml` n'est trouvé, l'installation de correctifs dépend des paramètres Inclusion de mises à jour non liées à la sécurité et Approbation automatique. Par exemple, si les mises à jour non liées à la sécurité sont autorisées, elles sont installées lorsque l'heure de l'approbation automatique arrive.

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 PatchFilt

Attribut	Description
	er de la référence de correctif. Indique le type de package inclus dans la notice de mise à jour. Vous pouvez consulter la liste des valeurs prises en charge à l'aide de la AWS CLI commande describe-patch-properties ou de l'opération APIDescribePatchProperties. Vous pouvez aussi afficher la liste dans la zone Règles d'approbation de la page Créer un référentiel de correctif ou la page Modifier le référentiel 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 consulter la liste des valeurs prises en charge à l'aide de la AWS CLI commande describe-patch-properties ou de l'opération APIDescribePatchProperties. Vous pouvez aussi afficher la liste dans la zone Règles d'approbation de la page Créer un référentiel de correctif ou la page Modifier le référentiel 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 ApprovedPatches 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 ApprovedPatches 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 afin de déterminer l'approbation des correctifs à des fins de déploiement.

> **Note**
>
> Pour obtenir des informations sur les formats acceptés de 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.

3. Le produit du nœud géré est déterminé par SSM Agent. 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
Référentiels de correctifs par défaut prédéfinis fournis par AWS et référentiels de correctifs personnalisés pour lesquels l'option Inclusion	Pour chaque notice de mise à jour dans `updateinfo.xml` , le référentiel de correctif est utilisée en guise de filtre, ce qui permet que seuls les packages qualifiés soient inclus

Option de sécurité	Sélection de correctifs
de mises à jour non liées à la sécuritén'est pas sélectionnée	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. Concernant les nœuds gérés de version 7, la commande YUM équivalente pour ce flux de travail est la suivante : `sudo yum update-minimal --sec-severity=Important,Moderate --bugfix -y` Concernant les nœuds gérés de version 8 et 9, la commande DNF équivalente pour ce flux de travail est la suivante : `sudo dnf upgrade-minimal --security --sec-severity=Moderate --sec-severity=Important`

Option de sécurité	Sélection de correctifs
Référentiels de correctifs personnalisés pour lesquels l'option Inclusion de mises à jour non liées à la sécurité est sélectionnée avec une liste de GRAVITÉ `[Critical, Important]` et une liste de CLASSIFICATION `[Security, Bugfix]`	En plus de l'application des mises à jour de sécurité qui ont été sélectionnées à partir de `updateinfo.xml` , Patch Manager applique les mises à jour non liées à la sécurité qui remplissent, par ailleurs, les règles de filtrage des correctifs. Concernant les nœuds gérés de version 7, la commande YUM équivalente pour ce flux de travail est la suivante : `sudo yum update --security --sec-severity=Critical,Important --bugfix -y` Concernant les nœuds gérés de version 8 et 9, la commande DNF équivalente pour ce flux de travail est la suivante : `sudo dnf upgrade --security --sec-severity=Critical, --sec-severity=Important --bugfix y`

Pour de plus amples informations sur les valeurs du statut de conformité des correctifs, veuillez consulter Comprendre les valeurs d'état de conformité des correctifs.

Comment fonctionnent les règles de base des correctifs sur AlmaLinuxRHEL, et Rocky Linux

Sur AlmaLinux, Red Hat Enterprise Linux (RHEL) etRocky Linux, le processus de sélection des correctifs est le suivant :

1. Sur le nœud géré, la bibliothèque YUM (RHEL7) ou la bibliothèque DNF (AlmaLinux 8 et 9, RHEL 8 et 9, et Rocky Linux 8 et 9) accède au `updateinfo.xml` fichier pour chaque dépôt 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 consulter la liste des valeurs prises en charge à l'aide de la AWS CLI commande describe-patch-properties ou de l'opération APIDescribePatchProperties. Vous pouvez aussi afficher la liste dans la zone Règles d'approbation de la page Créer un référentiel de correctif ou la page Modifier le référentiel 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 consulter la liste des valeurs prises en charge à l'aide de la AWS CLI commande describe-patch-properties ou de

Attribut	Description
	l'opération APIDescribePatchProperties. Vous pouvez aussi afficher la liste dans la zone Règles d'approbation de la page Créer un référentiel de correctif ou la page Modifier le référentiel de correctif dans la console Systems Manager.
update_id	Indique l'ID Advisory, tels que RHSA-2017 :0864. L'ID Advisory peut être utilisé dans l'attribut ApprovedPatches ou RejectedP atches, 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 ApprovedPatches 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 `ApproveAf terDays` est utilisée afin de détermine r l'approbation des correctifs à des fins de déploiement.

> **Note**
>
> Pour obtenir des informations sur les formats acceptés de 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.

3. Le produit du nœud géré est déterminé par SSM Agent. 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
Référentiels de correctifs par défaut prédéfinis fournis par AWS et référentiels de correctifs personnalisés pour lesquels la case Inclusion de mises à jour non liées à la sécurité n'est pas cochée	Pour chaque notice de mise à jour dans `updateinfo.xml` , le référentiel 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 AlmaLinux 8 et 9, RHEL 8 et 9, et Rocky Linux 8 et 9, la commande dnf équivalente pour ce flux de travail est la suivante : `sudo dnf upgrade-minimal --sec-severity=Critical --sec-severity=Important --bugfix -y`

Option de sécurité	Sélection de correctifs
Référentiels de correctifs personnalisés pour lesquels la case Inclusion de mises à jour non liées à la sécurité est cochée avec une liste de GRAVITÉ `[Critical, Important]` et une liste de CLASSIFICATION `[Security, Bugfix]`	En plus de l'application des mises à jour de sécurité qui ont été sélectionnées à partir de `updateinfo.xml` , Patch Manager applique les mises à jour non liées à la sécurité qui remplissent, 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 --sec-severity=Critical,Important --bugfix -y` Pour AlmaLinux 8 et 9, RHEL 8 et 9, et Rocky Linux 8 et 9, la commande dnf équivalente pour ce flux de travail est la suivante : `sudo dnf upgrade --sec-severity=Critical --sec-severity=Important --bugfix -y`

Pour de plus amples informations sur les valeurs du statut de conformité des correctifs, veuillez consulter Comprendre les valeurs d'état de conformité des correctifs.

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 Classification du type de données PatchFilter de la référence de correctifs. Indique le type de correctif inclus dans la notice de mise à jour.

  Vous pouvez consulter la liste des valeurs prises en charge à l'aide de la AWS CLI commande describe-patch-properties ou de l'opération APIDescribePatchProperties. Vous pouvez aussi

afficher la liste dans la zone Règles d'approbation de la page Créer un référentiel de correctif ou la page Modifier le référentiel de correctif dans la console Systems Manager.

- Severity : correspond à la valeur de l'attribut de la clé Severity dans le type de données PatchFilter de la référence de correctifs. Désigne la sévérité des correctifs.

  Vous pouvez consulter la liste des valeurs prises en charge à l'aide de la AWS CLI commande describe-patch-properties ou de l'opération APIDescribePatchProperties. Vous pouvez aussi afficher la liste dans la zone Règles d'approbation de la page Créer un référentiel de correctif ou la page Modifier le référentiel de correctif dans la console Systems Manager.

Le produit du nœud géré est déterminé par SSM Agent. 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, le référentiel 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 obtenir des informations sur les formats acceptés pour les listes de correctifs approuvés et de correctifs rejetés, veuillez consulter À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés.

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 le référentiel de correctif, Patch Manager 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. Si une mise à jour est disponible pour `python3-apt` (une interface de bibliothèque Python pour `libapt`), la mise à niveau est faite à la dernière version. (Ce package non lié à la sécurité est mis à niveau même si vous n'avez pas sélectionné l'option Inclure les mises à jour non liées à la sécurité.)

3. Ensuite, les listes GlobalFilters, ApprovalRules, ApprovedPatches 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.

Cependant, les règles d'approbation sont également assujetties au fait que la case Inclure les mises à jour non liées à la sécurité a été cochée ou non lors de la création ou de la dernière mise à jour d'un référentiel 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. Dans ce cas, pour Ubuntu Server les versions de correctifs candidates se limitent aux correctifs inclus dans les référentiels suivants :

- Ubuntu Server 14.04 LTS : `trusty-security`
- Ubuntu Server 16.04 LTS : `xenial-security`
- Ubuntu Server 18.04 LTS : `bionic-security`
- Ubuntu Server 20.04 LTS : `focal-security`
- Ubuntu Server 20.10 STR : `groovy-security`
- Ubuntu Server 22.04 LTS (`jammy-security`)
- Ubuntu Server23,04 () `lunar-security`

Si des mises à jour non liées à la sécurité sont incluses, les correctifs provenant d'autres référentiels sont également pris en compte.

> **Note**
>
> Pour obtenir des informations sur les formats acceptés pour les listes de correctifs approuvés et de correctifs rejetés, veuillez consulter À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés.

Pour afficher le contenu des champs Priorité et Section, exécutez la commande `aptitude` suivante :

> **Note**
>
> Il se peut que vous deviez d'abord 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 Comprendre les valeurs d'état de conformité des correctifs.

Différences clés entre l'application de correctifs dans Windows et Linux

Cette rubrique décrit les différences importantes entre l'application de correctifs sous Linux et WindowsPatch Manager, une fonctionnalité de AWS Systems Manager.

> **Note**
>
> Pour appliquer des correctifs à des nœuds gérés Linux, ces derniers doivent exécuter SSM Agent version 2.0.834.0 ou ultérieure.
> Une nouvelle version de SSM Agent est lancée chaque fois que de nouvelles fonctionnalités sont ajoutées à Systems Manager ou que des mises à jour sont apportées aux fonctionnalités existantes. Le fait de ne pas utiliser la dernière version de l'agent peut empêcher votre nœud géré d'utiliser diverses capacités et fonctionnalités de Systems Manager. C'est pourquoi nous vous recommandons d'automatiser le processus permettant de maintenir SSM Agent à jour sur vos machines. Pour plus d'informations, consultez Automatisation des mises à jour de l'SSM Agent. Abonnez-vous à la page des notes de SSM Agent publication GitHub pour recevoir des notifications concernant les SSM Agent mises à jour.

Différence 1 : Évaluation des correctifs

Linux

Pour l'application de correctifs Linux, Systems Manager vérifie les règles du référentiel de correctifs ainsi que la liste des correctifs approuvés et rejetés sur chaque nœud géré. Systems Manager doit vérifier l'application des correctifs sur chaque nœud, car le service récupère la liste des correctifs et mises à jour connus à partir des référentiels configurés sur le nœud géré.

Windows

Patch Manager utilise des processus différents sur les nœuds gérés Windows et Linux pour déterminer quels correctifs doivent être présents. Pour l'application de correctifs Windows, Systems Manager évalue les règles de référentiels 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 : Correctifs **`Not Applicable`**

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. Par exemple, un correctif `Not Applicable` est un correctif pour le logiciel Apache lorsqu'Apache n'est pas installé sur l'instance. Systems Manager indique le nombre de `Not Applicable` correctifs dans le résumé, mais si vous appelez l'DescribeInstancePatchesAPI d'un nœud géré, les données renvoyées n'incluent pas les correctifs dont l'état est égal à`Not Applicable`. Depuis Windows, ce comportement est différent.

Différence 3 : Prise en charge des documents SSM

Le document Systems Manager `AWS-ApplyPatchBaseline` (document SSM) ne prend pas en charge les nœuds gérés Linux. Pour appliquer des référentiels de correctifs à des nœuds gérés Linux, macOS et Windows Server, le document SSM recommandé est `AWS-RunPatchBaseline`. Pour plus d'informations, consultez À propos des documents SSM pour l'application de correctifs aux nœuds gérés et À propos du document SSM `AWS-RunPatchBaseline`.

Différence 4 : Correctifs d'applications

Le premier objectif de Patch Manager est d'appliquer des correctifs aux systèmes d'exploitation. Cependant, vous pouvez également utiliser Patch Manager pour appliquer des correctifs à certaines applications sur vos nœuds gérés.

Linux

Sur les systèmes d'exploitation Linux, Patch Manager utilise les référentiels configurés pour les mises à jour et ne fait pas la différence entre les correctifs de systèmes d'exploitation et les

correctifs d'applications. Vous pouvez utiliser Patch Manager pour définir les référentiels à partir desquels extraire les mises à jour. Pour plus d'informations, consultez Spécification d'un autre référentiel source de correctifs (Linux).

Windows

Sur les nœuds gérés Windows Server, vous pouvez appliquer des règles d'approbation, ainsi que les exceptions de correctif Approved (Approuvé) et Rejected (Rejeté) pour les applications publiées par Microsoft, telles que Microsoft Word 2016 et Microsoft Exchange Server 2016. Pour plus d'informations, voir Utilisation des référentiels de correctifs personnalisés.

À propos des documents SSM pour l'application de correctifs aux nœuds gérés

Cette rubrique décrit les neuf documents Systems Manager (documents SSM) actuellement disponibles pour vous aider à appliquer les dernières mises à jour de sécurité à vos nœuds gérés.

Nous vous recommandons d'utiliser cinq de ces documents seulement pour vos opérations d'application de correctifs. Conjointement, ces cinq documents SSM vous fournissent une gamme complète d'options de correctifs à l'aide d' AWS Systems Manager. Quatre de ces documents ont été publiés plus tard que les quatre documents SSM existants qu'ils remplacent et représentent les développements ou consolidations de fonctionnalités.

Documents SSM recommandés pour l'application de correctifs

Nous vous recommandons d'utiliser les cinq documents SSM suivants dans le cadre de vos opérations d'application de correctifs.

- `AWS-ConfigureWindowsUpdate`
- `AWS-InstallWindowsUpdates`
- `AWS-RunPatchBaseline`
- `AWS-RunPatchBaselineAssociation`
- `AWS-RunPatchBaselineWithHooks`

Documents SSM existants pour l'application de correctifs

Les quatre anciens documents SSM suivants peuvent toujours être utilisés dans certains cas, Régions AWS mais ils ne sont plus mis à jour, leur fonctionnement n'est pas garanti dans tous les

scénarios et pourrait ne plus être pris en charge à l'avenir. Nous vous recommandons de ne pas les utiliser dans le cadre de vos opérations d'application de correctifs.

- `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 l'application de correctifs aux nœuds gérés

L'utilisation des cinq documents SSM suivants est recommandée pour l'application de correctifs aux nœuds gérés.

Documents SSM recommandés

- AWS-ConfigureWindowsUpdate
- AWS-InstallWindowsUpdates
- AWS-RunPatchBaseline
- AWS-RunPatchBaselineAssociation
- AWS-RunPatchBaselineWithHooks

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 les nœuds gérés, selon les besoins. Utilisez ce document avec State Manager une fonctionnalité de AWS Systems Manager, pour vous assurer que Windows Update conserve sa configuration. Vous pouvez également l'exécuter manuellement en utilisant Run Command, une des fonctionnalités de AWS Systems Manager, 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 :

- Aucun

AWS-InstallWindowsUpdates

Installe les mises à jour sur un nœud géré Windows Server. Disponible dans toutes les Régions AWS.

Ce document SSM fournit des fonctionnalités de correctifs de base pour les cas suivants : lorsque vous souhaitez installer une mise à jour spécifique (à l'aide du paramètre `Include Kbs`) ou installer des correctifs avec les classifications ou catégories spécifiques, mais si vous 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 hérités pour l'application de correctifs aux nœuds gérés.

`AWS-RunPatchBaseline`

Installe des correctifs sur vos nœuds gérés ou analyse les nœuds pour 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 du référentiel de correctifs spécifié « par défaut » pour un type de système d'exploitation. informations de 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 de vos nœuds gérés en matière de correctifs, comme les nœuds auxquels il manque des correctifs et la nature de ces correctifs. Lorsque vous utilisez `AWS-RunPatchBaseline`, les informations de conformité des correctifs sont enregistrées à l'aide de la commande d'API `PutInventory`. Pour les systèmes d'exploitation Linux, des informations de conformité sont fournies sur les correctifs provenant à la fois du référentiel source par défaut configuré sur un nœud géré et de tout autre référentiel source spécifié par vos soins dans un référentiel de correctifs personnalisé. Pour en savoir plus sur les autres référentiels source, consultez Spécification d'un autre référentiel source de correctifs (Linux). Pour plus d'informations sur les outils de conformité Systems Manager, consultez Conformité d'AWS Systems Manager.

Remplace les documents existants :

- `AWS-ApplyPatchBaseline`

Le document hérité `AWS-ApplyPatchBaseline` s'applique uniquement aux nœuds gérés Windows Server et ne prévoit pas de prise en charge des correctifs d'application. 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 SSM Agent 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`.

AWS-RunPatchBaselineAssociation

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.

Des différences importantes distinguent `AWS-RunPatchBaselineAssociation` de `AWS-RunPatchBaseline` :

- `AWS-RunPatchBaselineAssociation`est destiné à être utilisé principalement avec des State Manager associations créées à l'aide Quick Setup d'une fonctionnalité de AWS Systems Manager. Précisément, lorsque vous utilisez le type de configuration de gestion des hôtes Quick Setup, si vous sélectionnez l'option Scan instances for missing patches daily (Analyser quotidiennement les instances pour les correctifs manquants), le système utilise `AWS-RunPatchBaselineAssociation` pour l'opération.

  Dans la plupart des cas, cependant, lors de la configuration de vos propres opérations d'application de correctifs, sélectionnez `AWS-RunPatchBaseline` ou `AWS-RunPatchBaselineWithHooks` de préférence à `AWS-RunPatchBaselineAssociation`.

  Pour plus d'informations, consultez les rubriques suivantes :

  - AWS Systems Manager Quick Setup
  - À propos du document SSM `AWS-RunPatchBaselineAssociation`
- `AWS-RunPatchBaselineAssociation` prend en charge l'utilisation de balises pour identifier le référentiel de correctifs à utiliser avec un ensemble de cibles lors de son exécution.
- Pour les opérations d'application de correctifs qui utilisent `AWS-RunPatchBaselineAssociation`, les données sur la conformité des correctifs sont compilées en fonction d'une association State Manager particulière. Les données sur la conformité des correctifs collectées lorsque `AWS-RunPatchBaselineAssociations`'exécute sont enregistrées avec la commande d'API `PutComplianceItems` plutôt qu'avec `PutInventory`. De cette façon, les données de conformité qui ne sont pas associées à cette association particulière ne sont pas écrasées.

  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 un référentiel de correctifs personnalisée. Pour en savoir plus sur les autres référentiels source, consultez Spécification d'un autre référentiel source de correctifs (Linux). Pour plus d'informations sur les outils de conformité Systems Manager, consultez Conformité d'AWS Systems Manager.

Remplace les documents existants :

- Aucun

Pour plus d'informations sur le document SSM `AWS-RunPatchBaselineAssociation`, consultez À propos du document SSM `AWS-RunPatchBaselineAssociation`.

`AWS-RunPatchBaselineWithHooks`

Installe les correctifs sur vos nœuds gérés, ou analyse les nœuds afin de déterminer s'il manque des correctifs qualifiés, avec des hooks facultatifs que vous pouvez utiliser pour exécuter des documents SSM à trois stades du cycle d'application des correctifs. Disponible dans toutes les Régions AWS commerciales.

`AWS-RunPatchBaselineWithHooks` diffère de `AWS-RunPatchBaseline` par son opération `Install`.

`AWS-RunPatchBaselineWithHooks` prend en charge les hooks de cycle de vie qui s'exécutent aux stades désignés lors de l'application de correctifs sur des nœuds gérés. Comme l'installation des correctifs exige parfois le redémarrage des nœuds gérés, l'opération d'application des correctifs se décompose en deux événements, pour un total de trois hooks qui prennent en charge la fonctionnalité personnalisée. Le premier hook précède l'opération `Install with NoReboot`. Le deuxième hook suit l'opération `Install with NoReboot`. Le troisième hook est disponible après le redémarrage du nœud.

Remplace les documents existants :

- Aucun

Pour plus d'informations sur le document SSM `AWS-RunPatchBaselineWithHooks`, consultez À propos du document SSM `AWS-RunPatchBaselineWithHooks`.

Documents SSM hérités pour l'application de correctifs aux nœuds gérés

Les quatre documents SSM suivants sont toujours disponibles dans certains Régions AWS cas. Cependant, ils ne sont plus mis à jour et pourraient ne plus être pris en charge à l'avenir. Nous ne recommandons donc pas leur utilisation. Utilisez plutôt les documents décrits dans Documents SSM recommandés pour l'application de correctifs aux nœuds gérés.

Documents SSM existants

- AWS-ApplyPatchBaseline
- AWS-FindWindowsUpdates
- AWS-InstallMissingWindowsUpdates
- AWS-InstallSpecificWindowsUpdates

AWS-ApplyPatchBaseline

Prend uniquement en charge les nœuds gérés Windows Server, mais n'inclut pas la prise en charge des correctifs d'application figurant dans le référentiel de remplacement, `AWS-RunPatchBaseline`. Non disponible en cas de Régions AWS lancement après août 2017.

> **Note**
>
> Le document de remplacement de ce document SSM, `AWS-RunPatchBaseline` nécessite une version 2.0.834.0 ou une version ultérieure de SSM Agent. Vous pouvez utiliser le document `AWS-UpdateSSMAgent` pour procéder à la mise à jour des nœuds gérés vers la dernière version de l'agent.

AWS-FindWindowsUpdates

Remplacé par `AWS-InstallWindowsUpdates`, qui peut effectuer toutes les mêmes actions. Non disponible en cas de Régions AWS lancement 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 versions 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 versions 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` = *`liste d'articles de la base de connaissances séparés par une virgule`*

À propos du document SSM **`AWS-RunPatchBaseline`**

AWS Systems Manager prend en charge`AWS-RunPatchBaseline`, un document Systems Manager (document SSM) pourPatch Manager, une fonctionnalité de AWS Systems Manager. Ce document SSM permet d'appliquer des correctifs sur les nœuds gérés, tant pour les mises à jour liées à la sécurité que pour les autres types de mises à jour. Si aucun groupe de correctifs n'est spécifié, lorsque le document est exécuté, il utilise le référentiel de correctifs spécifié « par défaut » pour un type de système d'exploitation. Sinon, il utilise le référentiel de correctifs associé au groupe de correctifs. Pour de plus amples informations sur les groupes de correctifs, veuillez consulter À propos des groupes de correctifs.

Vous pouvez utiliser le document `AWS-RunPatchBaseline` 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 publiées par Microsoft.)

Ce document prend en charge les nœuds gérés Linux, macOS et Windows Server. Ce document effectue les actions correspondant à chaque plateforme.

> **Note**
>
> Patch Manager prend également en charge le document SSM `AWS-ApplyPatchBaseline` hérité. Toutefois, seule l'application de correctifs sur les nœuds gérés Windows est prise en charge par ce document. Nous vous recommandons vivement de privilégier l'utilisation d'`AWS-RunPatchBaseline` dans la mesure où celui-ci prend en charge l'application de correctifs sur les nœuds gérés Linux, macOS et Windows Server. La version 2.0.834.0 ou une version ultérieure de SSM Agent pour pouvoir utiliser le document `AWS-RunPatchBaseline`.

Windows Server

Sur les nœuds Windows Server gérés, le `AWS-RunPatchBaseline` document télécharge et invoque un PowerShell module, qui télécharge à son tour un instantané de la ligne de base des correctifs qui s'applique au nœud géré. Cet instantané de référentiel de correctifs contient une liste des correctifs approuvés qui sont compilés en interrogeant le référentiel de correctifs sur un serveur Windows Server Update Services (WSUS). Cet instantané du référentiel de correctifs est transmis à l'API Windows Update qui contrôle le téléchargement et l'installation des correctifs approuvés selon les besoins.

Linux

Sur les nœuds gérés Linux, le document `AWS-RunPatchBaseline` appelle un module Python qui, à son tour, télécharge un instantané du référentiel de correctifs qui s'applique au nœud géré. Cet instantané du référentiel de correctifs utilise les règles définies et les listes de correctifs approuvés et bloqués afin de piloter le gestionnaire de package approprié pour chaque type de nœud :

- Les nœuds gérés Amazon Linux 1, Amazon Linux 2Oracle Linux, CentOS et RHEL 7 utilisent YUM. Pour les opérations YUM, Patch Manager nécessite `Python 2.6` ou une version ultérieure prise en charge (2.6 à 3.10).
- Les nœuds gérés RHEL 8 utilisent DNF. Pour les opérations DNF, Patch Manager nécessite une version prise en charge de `Python 2` ou `Python 3` (2.6 à 3.10). (Aucune des versions n'est installée par défaut sur RHEL 8. Vous devez les installer manuellement.)
- Les instances Debian Server, Raspberry Pi OS et Ubuntu Server utilisent APT. Pour les opérations APT, Patch Manager nécessite une version prise en charge de `Python 3` (3.0 à 3.10).

- Les nœuds gérés SUSE Linux Enterprise Server utilisent Zypper. Pour les opérations Zypper, Patch Manager nécessite `Python 2.6` ou une version ultérieure prise en charge (2.6 à 3.10).

macOS

Sur les nœuds gérés macOS, le document `AWS-RunPatchBaseline` appelle un module Python qui, à son tour, télécharge un instantané du référentiel de correctifs qui s'applique au nœud. Ensuite, un sous-processus Python invoque le AWS Command Line Interface (AWS CLI) sur le nœud pour récupérer les informations d'installation et de mise à jour pour les gestionnaires de packages spécifiés et pour piloter le gestionnaire de packages approprié pour chaque package de mise à jour.

Chaque instantané est spécifique à un groupe de correctifs Compte AWS, à un système d'exploitation et à un identifiant de capture d'écran. L'instantané est délivré via une URL Amazon Simple Storage Service (Amazon S3) présignée, qui expire 24 heures après la création de l'instantané. Toutefois, après l'expiration de l'URL, si vous souhaitez appliquer le même contenu d'instantané à d'autres nœuds gérés, générez une nouvelle URL Amazon S3 présignée jusqu'à trois jours après la création de l'instantané. Pour ce faire, utilisez la commande get-deployable-patch-snapshot-for-instance.

Une fois que toutes les mises à jour approuvées et applicables ont été installées, et que les redémarrages nécessaires ont été effectués, des informations relatives à la conformité des correctifs sont générées sur un nœud géré et transmises à Patch Manager.

> **Note**
>
> Si le paramètre `RebootOption` est défini sur `NoReboot` dans le document `AWS-RunPatchBaseline`, le nœud géré n'est pas redémarré après l'exécution de Patch Manager. Pour plus d'informations, consultez Nom du paramètre: `RebootOption`.

Pour plus d'informations sur l'affichage des données de conformité des correctifs, consultez A propos de la conformité des correctifs.

`AWS-RunPatchBaseline` paramètres

`AWS-RunPatchBaseline` prend en charge cinq paramètres. Le paramètre `Operation` est obligatoire. Les paramètres `InstallOverrideList`, `BaselineOverride` et `RebootOption`

sont facultatifs. `Snapshot-ID` est techniquement facultatif, mais nous vous recommandons de lui attribuer une valeur personnalisée lorsque vous exécutez `AWS-RunPatchBaseline`en dehors d'une fenêtre de maintenance, et de laisser Patch Manager fournir automatiquement la valeur personnalisée lorsque le document est exécuté dans le cadre d'une opération de fenêtre de maintenance.

Paramètres

- Nom du paramètre: Operation
- Nom du paramètre: AssociationId
- Nom du paramètre: Snapshot ID
- Nom du paramètre: InstallOverrideList
- Nom du paramètre: RebootOption
- Nom du paramètre: BaselineOverride

Nom du paramètre: **`Operation`**

Utilisation : Obligatoire.

Options : `Scan` | `Install`.

Analyser

Lorsque vous sélectionnez l'option `Scan`, `AWS-RunPatchBaseline` détermine l'état de conformité du nœud géré en matière de correctifs et transmet cette information à Patch Manager. `Scan` n'invite pas à installer les mises à jour ou à redémarrer les nœuds gérés. Mais l'opération identifie les mises à jour manquantes qui sont approuvées et applicables au nœud.

Installation

Lorsque vous sélectionnez l'option `Install`, `AWS-RunPatchBaseline` tente d'installer les mises à jour approuvées et applicables qu'il manque sur le nœud géré. Les informations de 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. Chaque fois qu'une mise à jour est installée sur un nœud géré, ce dernier est redémarré 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`, le nœud

géré n'est pas redémarré après l'exécution de Patch Manager. Pour plus d'informations, consultez Nom du paramètre: `RebootOption`).

> **Note**
>
> Si un correctif spécifié par les règles de référentiel est installé avant la mise à jour du nœud géré par Patch Manager, 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: **`AssociationId`**

Utilisation : Facultatif.

`AssociationId` est l'ID d'une association existante dans State Manager, une fonctionnalité de AWS Systems Manager. Il est utilisé par Patch Manager pour ajouter des données de conformité à une association spécifiée. Cette association est liée à une opération d'application de correctifs définie dans une politique de correctifs dans Quick Setup.

> **Note**
>
> Avec le `AWS-RunPatchBaseline`, si une valeur `AssociationId` est fournie avec un remplacement du référentiel de la politique de correctifs, l'application des correctifs est effectuée en tant qu'opération `PatchPolicy` et la valeur `ExecutionType` indiquée dans `AWS:ComplianceItem` est également `PatchPolicy`. Si aucune valeur `AssociationId` n'est fournie, l'application des correctifs est effectuée en tant qu'opération `Command` et le rapport de valeur `ExecutionType` sur l'`AWS:ComplianceItem` soumis est également `Command`.

Si vous n'avez pas encore d'association à utiliser, vous pouvez en créer une en exécutant la commande create-association.

Nom du paramètre: **`Snapshot ID`**

Utilisation : Facultatif.

`Snapshot ID` est un ID unique (GUID) utilisé par Patch Manager pour s'assurer que les nœuds gérés d'un groupe auquel des correctifs ont été appliqués dans le cadre d'une opération individuelle disposent tous du 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	Détails
Exécution de `AWS-RunPatchBaseline` à l'intérieur d'une fenêtre de maintenance	Ne fournissez pas d'ID d'instantané. Patch Manager le fournira pour vous.	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 pendant plus de trois jours, l'instantané du référentiel de correctifs peut changer. Par la suite, un nouvel instantané est généré même si vous spécifiez le même ID après l'expiration de l'instantané.

Mode	Bonne pratique	Détails
Exécution de `AWS-RunPatchBaseline` à l'extérieur d'une fenêtre de maintenance	Générez et spécifiez une valeur de GUID personnalisée pour l'ID d'instantané.[1]	Si vous n'avez pas recours à une fenêtre de maintenance pour exécuter `AWS-RunPatchBaseline`, nous vous recommandons de générer et de spécifier un ID d'instantané unique pour chaque référentiel de correctifs, en particulier si vous exécutez le document `AWS-RunPatchBaseline` sur plusieurs nœuds gérés au cours de la même opération. Dans ce cas de figure, si vous ne spécifiez pas d'ID, Systems Manager génère un ID d'instantané différent pour chacun des nœuds gérés auxquels la commande est envoyée. Cela peut entraîner la spécification de différents ensembles de correctifs parmi les nœuds gérés. Par exemple, si vous exécutez le document `AWS-RunPatchBaseline` directement via la fonctionnalité Run Command d' AWS Systems Manager et que vous ciblez un groupe de 50 nœuds gérés. La spécification d'un ID d'instantané personnalisé entraîne la génération d'un instantané de référentiel unique qui permet d'évaluer

Mode	Bonne pratique	Détails
		et de corriger tous les nœuds, garantissant ainsi un état final 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 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 le référentiel de correctifs par défaut actuelle. Cela vous confère un contrôle plus précis sur les correctifs installés sur vos nœuds gérés.

Le comportement de l'opération de correction lors de l'utilisation du `InstallOverrideList` paramètre diffère entre les nœuds Linux et macOS gérés et les nœuds Windows Server gérés. Sur Linux &macOS, Patch Manager tente d'appliquer les correctifs inclus dans la liste des `InstallOverrideList` correctifs présents dans n'importe quel référentiel activé sur le nœud, que les correctifs respectent ou non les règles de base des correctifs. Sur Windows Server les nœuds, toutefois, les correctifs de la liste des `InstallOverrideList` correctifs ne sont appliqués que s'ils correspondent également aux règles de base des correctifs.

Sachez que les rapports de conformité reflètent les états de correctif en fonction de ce qui est spécifié dans le référentiel de correctifs et non pas de ce que vous spécifiez dans une liste `InstallOverrideList` de correctifs. En d'autres termes, les opérations d'analyse ignorent le paramètre `InstallOverrideList`. Cela permet de garantir que les rapports de conformité reflètent constamment les états de correctif en fonction de la politique 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 paramètre InstallOverrideList dans AWS-RunPatchBaseline ou AWS-RunPatchBaselineAssociation.

Formats d'URL valides

> **Note**
>
> Si votre fichier est stocké dans un compartiment accessible au public, vous pouvez spécifier un format d'URL https ou une URL de style chemin Amazon S3. Si votre fichier est stocké dans un compartiment privé, vous devez spécifier une URL de style chemin Amazon S3.

- Format des URL https :

```
https://s3.aws-api-domain/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 les correctifs dans votre liste dépendent du système d'exploitation de votre nœud géré. 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. Par exemple : `'dhclient.x86_64'`. Vous pouvez utiliser des caractères génériques dans l'ID pour indiquer plusieurs packages. Par 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. Par exemple : `'*32:9.8.2-0.*.rc1.57.amzn1'`.

Par exemple :

- La version du package apt qui est actuellement installée sur votre nœud géré est la version 1.2.25, 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, l'installation de la version 1.2.27 d'apt sera bloquée et signalée comme « Failed-NonCompliant ».

Windows Server

id

Le champ id est obligatoire. Utilisez-le pour spécifier des correctifs à l'aide des ID de la base de connaissance Microsoft (par exemple, KB2736693) et des ID de bulletins de sécurité Microsoft (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.

macOS

id

Le champ id est obligatoire. La valeur du champ id peut être fournie sous un format `{package-name}.{package-version}` ou un format {package_name}.

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'
```

- macOS

```
patches:
    -
        id: 'XProtectPlistConfigData'
    -
        id: 'MRTConfigData.1.61'
    -
        id: 'Command Line Tools for Xcode.11.5'
    -
        id: 'Gatekeeper Configuration Data'
```

- 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`

Par défaut : `RebootIfNeeded`

> **Warning**
>
> L'option par défaut est `RebootIfNeeded`. Veillez à sélectionner l'option qui correspond à votre cas d'utilisation. Par exemple, si vos nœuds gérés doivent redémarrer immédiatement pour finaliser un processus de configuration, sélectionnez `RebootIfNeeded`. Ou, si des nœuds gérés doivent rester disponibles jusqu'à une heure de redémarrage planifiée, sélectionnez `NoReboot`.

> **Important**
>
> Nous vous déconseillons de Patch Manager les utiliser pour appliquer des correctifs à des instances de cluster dans Amazon EMR (précédemment appelé Amazon MapReduce Elastic). Ne sélectionnez pas l'option `RebootIfNeeded` pour le paramètre `RebootOption`. (Cette option est disponible dans les documents SSM Command pour l'application de correctifs sur `AWS-RunPatchBaseline`, `AWS-RunPatchBaselineAssociation` et `AWS-RunPatchBaselineWithHooks`.)
>
> Les commandes sous-jacentes pour l'application de correctifs à l'aide de Patch Manager utilisent les commandes `yum` et `dnf`. Par conséquent, les opérations entraînent des incompatibilités en raison de la manière dont les packages sont installés. Pour plus d'informations sur les méthodes préférées de mise à jour logicielle sur les clusters Amazon EMR, veuillez consulter la rubrique Utilisation de l'AMI par défaut pour Amazon EMR dans le Guide de gestion Amazon EMR.

RebootIfNeeded

Lorsque vous sélectionnez l'option `RebootIfNeeded`, le nœud géré est redémarré dans l'un des cas suivants :

- Patch Manager a installé un ou plusieurs correctifs.

  Patch Manager n'évalue pas si un redémarrage est requis par le correctif. Le système est redémarré même si le correctif ne nécessite pas de redémarrage.

- Patch Manager détecte un ou plusieurs correctifs à l'état `INSTALLED_PENDING_REBOOT` durant l'opération `Install`.

  L'`INSTALLED_PENDING_REBOOT`état peut indiquer que l'option `NoReboot` a été sélectionnée lors de la dernière exécution de l'`Install`opération ou qu'un correctif a été installé en dehors de cette zone Patch Manager depuis le dernier redémarrage du nœud géré.

Dans ces deux cas, le redémarrage des nœuds gérés permet de supprimer les packages mis à jour de la mémoire, et assure la cohérence du comportement d'application des correctifs et de redémarrage sur tous les systèmes d'exploitation.

NoReboot

Lorsque vous sélectionnez l'option `NoReboot`, Patch Manager ne redémarre pas le nœud géré même s'il y a installé des correctifs pendant l'opération `Install`. Cette option est utile si vous savez qu'il n'est pas nécessaire de redémarrer vos nœuds gérés après l'application de correctifs, ou si des applications ou des processus en cours d'exécution sur un nœud ne doivent pas être perturbés par un redémarrage suite à l'application de correctifs. Elle est également utile lorsque vous souhaitez bénéficier de plus de contrôle sur le timing des redémarrages des nœuds gérés, par exemple en utilisant une fenêtre de maintenance.

> **Note**
>
> Si vous sélectionnez l'option `NoReboot` et qu'un correctif est installé, l'état du correctif est attribué au correctif `InstalledPendingReboot`. Le nœud géré, quant à lui, est marqué comme `Non-Compliant`. Après un redémarrage et l'exécution d'une opération `Scan`, l'état du nœud géré est mis à jour et devient `Compliant`.

Fichier de suivi de l'installation des correctifs : pour suivre l'installation des correctifs, en particulier ceux installés depuis le dernier redémarrage du système, Systems Manager gère un fichier sur le nœud géré.

> **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 correspondant au nœud géré est inexact. Dans ce cas, redémarrez le nœud et lancez une opération d'analyse des correctifs pour restaurer le fichier.

Ce fichier de suivi est stocké aux emplacements suivants sur vos nœuds gérés :

- Systèmes d'exploitation Linux :
  - `/var/log/amazon/ssm/patch-configuration/patch-states-configuration.json`
  - `/var/log/amazon/ssm/patch-configuration/patch-inventory-from-last-operation.json`
- Système d'exploitation Windows Server :
  - `C:\ProgramData\Amazon\PatchBaselineOperations\State\PatchStatesConfiguration.json`
  - `C:\ProgramData\Amazon\PatchBaselineOperations\State\PatchInventoryFromLastOperation.json`

Nom du paramètre: **`BaselineOverride`**

Utilisation : Facultatif.

Vous pouvez définir des préférences d'application de correctifs au moment de l'exécution en utilisant le paramètre `BaselineOverride`. Ce remplacement de référentiel est conservé en tant qu'objet JSON dans un compartiment S3. Il garantit que les opérations d'application de correctifs utilisent les référentiels correspondant au système d'exploitation hôte au lieu d'appliquer les règles du référentiel de correctifs par défaut

Pour de plus amples informations sur l'utilisation du paramètre `BaselineOverride`, veuillez consulter Utilisation du BaselineOverride paramètre.

À propos du document SSM **AWS-RunPatchBaselineAssociation**

Comme le document AWS-RunPatchBaseline, AWS-RunPatchBaselineAssociation effectue des opérations d'application de correctifs, pour les mises à jour liées à la sécurité et pour 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 publiées par Microsoft.)

Ce document prend en charge les instances Amazon Elastic Compute Cloud (Amazon EC2) pour Linux, macOS et Windows Server. Il ne prend pas en charge les nœuds non EC2 d'un environnement hybride et multicloud. Le document exécutera les actions appropriées pour chaque plate-forme, en invoquant un module Python sur Linux et les macOS instances, et un PowerShell module sur les instances Windows.

AWS-RunPatchBaselineAssociation diffère cependant de AWS-RunPatchBaseline par les aspects suivants :

- AWS-RunPatchBaselineAssociation est principalement destinée à une utilisation avec les associations State Manager créées avec Quick Setup, une fonctionnalité d' AWS Systems Manager. Précisément, lorsque vous utilisez le type de configuration de gestion des hôtes Quick Setup, si vous sélectionnez l'option Scan instances for missing patches daily (Analyser quotidiennement les instances pour les correctifs manquants), le système utilise AWS-RunPatchBaselineAssociation pour l'opération.

  Dans la plupart des cas, cependant, lors de la configuration de vos propres opérations d'application de correctifs, sélectionnez AWS-RunPatchBaseline ou AWS-RunPatchBaselineWithHooks de préférence à AWS-RunPatchBaselineAssociation.
- Lorsque vous utilisez le document AWS-RunPatchBaselineAssociation, vous pouvez spécifier une paire clé-balise dans le champ de paramètre BaselineTags du document. Si une ligne de base de correctifs personnalisée Compte AWS partage ces balises, une fonctionnalité de Patch Manager AWS Systems Manager, utilise cette ligne de base balisée lorsqu'elle s'exécute sur les instances cibles au lieu de la ligne de base de correctifs « par défaut » actuellement spécifiée pour le type de système d'exploitation.

> **Important**
>
> Si vous choisissez d'utiliser `AWS-RunPatchBaselineAssociation` dans des opérations d'application de correctifs autres que celles configurées avec Quick Setup, et que vous voulez utiliser son paramètre facultatif `BaselineTags`, vous devez ajouter des autorisations supplémentaires au profil d'instance pour les instances Amazon Elastic Compute Cloud (Amazon EC2). Pour plus d'informations, consultez Nom du paramètre: `BaselineTags`.

Les deux formats suivants sont valides pour votre paramètre `BaselineTags` :

`Key=tag-key,Values=tag-value`

`Key=tag-key,Values=tag-value1,tag-value2,tag-value3`

- Lors de l'exécution de `AWS-RunPatchBaselineAssociation`, les données de conformité des correctifs qu'il collecte sont enregistrées à l'aide de la commande d'API `PutComplianceItems` au lieu de la commande `PutInventory`, qui est utilisée par `AWS-RunPatchBaseline`. Cette différence signifie que les informations de conformité des correctifs sont stockées et signalées pour une association particulière. Les données sur la conformité des correctifs générées hors de cette association ne sont pas remplacées.
- Les informations de conformité des correctifs signalées après l'exécution de `AWS-RunPatchBaselineAssociation` indiquent si une instance est conforme ou non. Il n'inclut pas les détails au niveau du correctif, comme le montre le résultat de la commande suivante AWS Command Line Interface (AWS CLI). La commande exécute un filtrage sur `Association` 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 retourne des informations telles que les suivantes.

```
{
    "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 clé-balise a été spécifiée comme paramètre pour le document `AWS-RunPatchBaselineAssociation`, Patch Manager recherche un référentiel de correctifs personnalisé correspondant au type de système d'exploitation et qui a été balisé avec la même paire clé-balise. Cette recherche ne se limite pas au référentiel de correctifs par défaut actuellement spécifié ou au référentiel affecté à un groupe de correctifs. Si aucun référentiel n'est trouvé avec les balises spécifiées, Patch Managerrecherche ensuite un groupe de correctifs, si un groupe a été spécifié dans la commande qui exécute `AWS-RunPatchBaselineAssociation`. Si aucun groupe de correctifs ne correspond, Patch Manager revient au référentiel de correctifs par défaut actuel pour le compte de système d'exploitation.

Si plusieurs référentiels de correctifs sont trouvés avec les balises spécifiées dans le document `AWS-RunPatchBaselineAssociation`, Patch Manager renvoie un message d'erreur indiquant qu'un seul référentiel de correctifs peut être balisé avec cette paire clé-valeur pour que l'opération continue.

> **Note**
>
> Sur les instances Linux, le gestionnaire de packages approprié pour chaque type d'instance est utilisé pour installer les packages :

- Amazon Linux 1, Amazon Linux 2, CentOS et les Oracle Linux RHEL instances utilisent YUM. Pour les opérations YUM, Patch Manager nécessite `Python 2.6` ou une version ultérieure prise en charge (2.6 à 3.10).
- Les instances Debian Server, Raspberry Pi OS et Ubuntu Server utilisent APT. Pour les opérations APT, Patch Manager nécessite une version prise en charge de `Python 3` (3.0 à 3.10).
- Les instances SUSE Linux Enterprise Server utilisent Zypper. Pour les opérations Zypper, Patch Manager nécessite `Python 2.6` ou une version ultérieure prise en charge (2.6 à 3.10).

Une fois l'analyse terminée ou toutes les mises à jour approuvées et applicables installées, et les redémarrages exécutés au besoin, les informations de conformité des correctifs sont générées sur une instance et rapportées au service Patch Compliance.

> **Note**
>
> Si le paramètre `RebootOption` est défini sur `NoReboot` dans le document `AWS-RunPatchBaselineAssociation`, l'instance n'est pas redémarrée après l'exécution du Patch Manager. Pour plus d'informations, consultez Nom du paramètre: `RebootOption`.

Pour plus d'informations sur l'affichage des données de conformité des correctifs, consultez A propos de la conformité des correctifs.

`AWS-RunPatchBaselineAssociation` paramètres

`AWS-RunPatchBaselineAssociation` prend en charge quatre paramètres. Les paramètres `Operation` et `AssociationId` sont obligatoires. Les paramètres `InstallOverrideList`, `RebootOption` et `BaselineTags` sont facultatifs.

Paramètres

- Nom du paramètre: Operation
- Nom du paramètre: BaselineTags
- Nom du paramètre: AssociationId
- Nom du paramètre: InstallOverrideList

- Nom du paramètre: RebootOption

Nom du paramètre: **`Operation`**

Utilisation : Obligatoire.

Options : `Scan` | `Install`.

Analyser

Lorsque vous sélectionnez l'option `Scan`, `AWS-RunPatchBaselineAssociation` détermine l'état de conformité de correctif de l'instance et rapporte cette information au Patch Manager. `Scan` n'invite pas à installer des mises à jour ou à redémarrer des instances. Cet opération identifie plutôt à quel emplacement il manque des mises à jour approuvées et applicables à l'instance.

Installation

Lorsque vous sélectionnez l'option `Install`, `AWS-RunPatchBaselineAssociation` tente d'installer les mises à jour approuvées et applicables qui sont manquantes dans l'instance. Les informations de 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-RunPatchBaselineAssociation`, l'instance n'est pas redémarrée après l'exécution de Patch Manager. Pour de plus amples informations, consultez Nom du paramètre: `RebootOption`.)

> **Note**
>
> Si un correctif spécifié par les règles de ligne de base est installé avant que Patch Manager ne mette à jour 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 sélectionnez et affectez à un référentiel de correctifs personnalisé. Vous pouvez spécifier une ou plusieurs valeurs pour ce paramètre. Les deux formats sont valides :

`Key=`*`tag-key`*`,Values=`*`tag-value`*

`Key=`*`tag-key`*`,Values=`*`tag-value1`*`,`*`tag-value2`*`,`*`tag-value3`*

La valeur `BaselineTags` est utilisée par Patch Manager pour s'assurer qu'un ensemble d'instances corrigées lors d'une opération unique dispose du même ensemble de correctifs identiques approuvés. Lorsque l'opération d'application de correctifs s'exécute, Patch Manager vérifie si un référentiel de correctifs pour le type de système d'exploitation est balisé avec la même paire clé-valeur que celle spécifiée pour `BaselineTags`. En cas de correspondance, ce référentiel de correctifs personnalisé est utilisé. En l'absence de correspondance, un référentiel de correctifs est identifié en fonction de n'importe quel groupe de correctifs spécifié pour l'application de correctifs. S'il n'y en a pas, la ligne de base de correctifs prédéfinie AWS gérée pour ce système d'exploitation est utilisée.

Exigences d'autorisations supplémentaires

Si vous utilisez `AWS-RunPatchBaselineAssociation` dans des opérations d'application de correctifs autres que celles configurées avec Quick Setup, et que vous voulez utiliser son paramètre facultatif `BaselineTags`, vous devez ajouter les autorisations suivantes au profil d'instance pour les instances Amazon Elastic Compute Cloud (Amazon EC2).

> **Note**
>
> Quick Setup et `AWS-RunPatchBaselineAssociation` ne prennent pas en charge les serveurs et les machines virtuelles (VM) sur site.

```
{
    "Effect": "Allow",
    "Action": [
        "ssm:DescribePatchBaselines",
        "tag:GetResources"
```

```
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "ssm:GetPatchBaseline",
        "ssm:DescribeEffectivePatchesForPatchBaseline"
    ],
    "Resource": "patch-baseline-arn"
}
```

Remplacez-le *patch-baseline-arn*par le Amazon Resource Name (ARN) de la ligne de base de correctif à laquelle vous souhaitez donner accès, au format`arn:aws:ssm:us-east-2:123456789012:patchbaseline/pb-0c10e65780EXAMPLE`.

Nom du paramètre: **`AssociationId`**

Utilisation : Obligatoire.

`AssociationId` est l'ID d'une association existante dans State Manager, une fonctionnalité de AWS Systems Manager. Il est utilisé par Patch Manager pour ajouter des données de conformité à une association spécifiée. Cette association est liée à une opération de correctif `Scan` activée dans une configuration de gestion des hôtes créée dans Quick Setup. En envoyant des résultats d'application de correctifs sous forme de données de conformité d'association plutôt que de données de conformité d'inventaire, les informations de conformité d'inventaire existantes pour vos instances ne sont pas remplacées après une opération d'application de correctifs, ni pour d'autres ID d'association. Si vous n'avez pas encore d'association à utiliser, vous pouvez en créer une en exécutant la commande create-association. Par exemple :

Linux & macOS

```
aws ssm create-association \
    --name "AWS-RunPatchBaselineAssociation" \
    --association-name "MyPatchHostConfigAssociation" \
    --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 Server

```
aws ssm create-association ^
    --name "AWS-RunPatchBaselineAssociation" ^
    --association-name "MyPatchHostConfigAssociation" ^
    --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.

En utilisant `InstallOverrideList`, vous spécifiez une URL https ou une URL de style chemin Amazon Simple Storage Service (Amazon S3) à une liste de correctifs à installer. Cette liste d'installation de correctifs que vous conservez au format YAML remplace les correctifs spécifiés par le référentiel de correctifs par défaut actuelle. Cela vous offre un contrôle plus précis sur les correctifs installés sur vos instances.

Le comportement de l'opération de correction lors de l'utilisation du `InstallOverrideList` paramètre diffère entre les nœuds Linux et macOS gérés et les nœuds Windows Server gérés. Sur Linux &macOS, Patch Manager tente d'appliquer les correctifs inclus dans la liste des `InstallOverrideList` correctifs présents dans n'importe quel référentiel activé sur le nœud, que les correctifs respectent ou non les règles de base des correctifs. Sur Windows Server les nœuds, toutefois, les correctifs de la liste des `InstallOverrideList` correctifs ne sont appliqués que s'ils correspondent également aux règles de base des correctifs.

Sachez que les rapports de conformité reflètent les états de correctif en fonction de ce qui est spécifié dans le référentiel de correctifs et non pas de ce que vous spécifiez dans une liste `InstallOverrideList` de correctifs. En d'autres termes, les opérations d'analyse ignorent le paramètre `InstallOverrideList`. Cela permet de garantir que les rapports de conformité reflètent constamment les états de correctif en fonction de la politique plutôt que de ce qui a été approuvé pour une opération spécifique d'application de correctifs.

Formats d'URL valides

> **Note**
>
> Si votre fichier est stocké dans un compartiment accessible au public, vous pouvez spécifier un format d'URL https ou une URL de style chemin Amazon S3. Si votre fichier est stocké dans un compartiment privé, vous devez spécifier une URL de style chemin Amazon S3.

- Exemple de format d'URL https :

```
https://s3.amazonaws.com/DOC-EXAMPLE-BUCKET/my-windows-override-list.yaml
```

- Exemple d'URL de type chemin d'accès 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 des ID de la base de connaissance Microsoft (par exemple, KB2736693) et des ID de bulletins de sécurité Microsoft (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. Par exemple : `'dhclient.x86_64'`. Vous pouvez utiliser des caractères génériques dans l'ID pour indiquer plusieurs packages. Par 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. Par exemple : `'*32:9.8.2-0.*.rc1.57.amzn1'`.

  Par 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, l'installation de la version 1.2.27 d'apt sera bloquée et signalée 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`

Par défaut : `RebootIfNeeded`

> **Warning**
>
> L'option par défaut est `RebootIfNeeded`. Veillez à sélectionner l'option qui correspond à votre cas d'utilisation. Par exemple, si un redémarrage immédiat de vos instances est nécessaire pour finaliser un processus de configuration, sélectionnez `RebootIfNeeded`. Ou, si des instances doivent rester disponibles jusqu'à une heure de redémarrage planifiée, sélectionnez `NoReboot`.

> **Important**
>
> Nous vous déconseillons de Patch Manager les utiliser pour appliquer des correctifs à des instances de cluster dans Amazon EMR (précédemment appelé Amazon MapReduce Elastic). Ne sélectionnez pas l'option `RebootIfNeeded` pour le paramètre `RebootOption`. (Cette option est disponible dans les documents SSM Command pour l'application de

correctifs sur `AWS-RunPatchBaseline`, `AWS-RunPatchBaselineAssociation` et `AWS-RunPatchBaselineWithHooks`.)
Les commandes sous-jacentes pour l'application de correctifs à l'aide de Patch Manager utilisent les commandes `yum` et `dnf`. Par conséquent, les opérations entraînent des incompatibilités en raison de la manière dont les packages sont installés. Pour plus d'informations sur les méthodes préférées de mise à jour logicielle sur les clusters Amazon EMR, veuillez consulter la rubrique Utilisation de l'AMI par défaut pour Amazon EMR dans le Guide de gestion Amazon EMR.

RebootIfNeeded

Lorsque vous sélectionnez l'option `RebootIfNeeded`, l'instance est redémarrée dans l'un des cas suivants :

- Patch Manager a installé un ou plusieurs correctifs.

  Patch Manager n'évalue pas si un redémarrage est requis par le correctif. Le système est redémarré même si le correctif ne nécessite pas de redémarrage.

- Patch Manager détecte un ou plusieurs correctifs à l'état `INSTALLED_PENDING_REBOOT` durant l'opération `Install`.

  L'`INSTALLED_PENDING_REBOOT`état peut indiquer que l'option `NoReboot` a été sélectionnée lors de la dernière exécution de l'`Install`opération ou qu'un correctif a été installé en dehors de cette zone Patch Manager depuis le dernier redémarrage du nœud géré.

Dans ces deux cas, le redémarrage des instances garantit que les packages mis à jour sont supprimés de la mémoire et assure la cohérence du comportement d'application de correctifs et de redémarrage sur tous les systèmes d'exploitation.

NoReboot

Lorsque vous sélectionnez l'option `NoReboot`, Patch Manager ne redémarre pas une instance même s'il a installé des correctifs pendant l'opération `Install`. 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 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`
  - `/var/log/amazon/ssm/patch-configuration/patch-inventory-from-last-operation.json`
- Système d'exploitation Windows Server :
  - `C:\ProgramData\Amazon\PatchBaselineOperations\State\PatchStatesConfiguration.json`
  - `C:\ProgramData\Amazon\PatchBaselineOperations\State\PatchInventoryFromLastOperation.json`

À propos du document SSM **`AWS-RunPatchBaselineWithHooks`**

AWS Systems Manager prend en charge`AWS-RunPatchBaselineWithHooks`, un document Systems Manager (document SSM) pourPatch Manager, une fonctionnalité de AWS Systems Manager. Ce document SSM permet d'appliquer des correctifs sur les nœuds gérés, tant pour les mises à jour liées à la sécurité que pour les autres types de mises à jour.

`AWS-RunPatchBaselineWithHooks` diffère de `AWS-RunPatchBaseline` par les aspects suivants :

- Un document wrapper : `AWS-RunPatchBaselineWithHooks` est un wrapper pour `AWS-RunPatchBaseline` et s'appuie sur `AWS-RunPatchBaseline` pour certaines de ses opérations.

- L'opération **`Install`** : `AWS-RunPatchBaselineWithHooks` prend en charge les hooks de cycle de vie qui s'exécutent aux stades désignés lors de l'application de correctifs sur des nœuds gérés. Comme l'installation des correctifs exige parfois le redémarrage des nœuds gérés, l'opération d'application des correctifs se décompose en deux événements, pour un total de trois hooks qui prennent en charge la fonctionnalité personnalisée. Le premier hook précède l'opération `Install with NoReboot`. Le deuxième hook suit l'opération `Install with NoReboot`. Le troisième hook est disponible après le redémarrage du nœud géré.
- Aucune prise en charge de liste de correctifs personnalisée : `AWS-RunPatchBaselineWithHooks` ne prend pas en charge le paramètre `InstallOverrideList`.
- Prise en charge de SSM Agent : `AWS-RunPatchBaselineWithHooks` exige que SSM Agent 3.0.502 (ou version ultérieure) soit installé sur le nœud géré auquel les correctifs doivent être appliqués.

Si aucun groupe de correctifs n'est spécifié, lorsque le document est exécuté, il utilise le référentiel de correctifs « par défaut » actuellement spécifié pour un type de système d'exploitation. Sinon, il utilise les référentiels de correctifs associés au groupe de correctifs. Pour de plus amples informations sur les groupes de correctifs, veuillez consulter À propos des groupes de correctifs.

Vous pouvez utiliser le document `AWS-RunPatchBaselineWithHooks` pour appliquer des correctifs pour les systèmes d'exploitation et les applications. (Sur Windows, la prise en charge des applications est limitée à des mises à jour pour les applications publiées par Microsoft.)

Ce document prend en charge les nœuds gérés Linux, macOS et Windows Server. Ce document effectue les actions correspondant à chaque plateforme.

Linux

Sur les nœuds gérés Linux, le document `AWS-RunPatchBaselineWithHooks` appelle un module Python qui, à son tour, télécharge un instantané du référentiel de correctifs qui s'applique au nœud géré. Cet instantané du référentiel de correctifs utilise les règles définies et les listes de correctifs approuvés et bloqués afin de piloter le gestionnaire de package approprié pour chaque type de nœud :

- Les nœuds gérés Amazon Linux 1, Amazon Linux 2Oracle Linux, CentOS et RHEL 7 utilisent YUM. Pour les opérations YUM, Patch Manager nécessite `Python 2.6` ou une version ultérieure prise en charge (2.6 à 3.10).

- Les nœuds gérés RHEL 8 utilisent DNF. Pour les opérations DNF, Patch Manager nécessite une version prise en charge de `Python 2` ou `Python 3` (2.6 à 3.10). (Aucune des versions n'est installée par défaut sur RHEL 8. Vous devez les installer manuellement.)
- Les instances Debian Server, Raspberry Pi OS et Ubuntu Server utilisent APT. Pour les opérations APT, Patch Manager nécessite une version prise en charge de `Python 3` (3.0 à 3.10).
- Les nœuds gérés SUSE Linux Enterprise Server utilisent Zypper. Pour les opérations Zypper, Patch Manager nécessite `Python 2.6` ou une version ultérieure prise en charge (2.6 à 3.10).

macOS

Sur les nœuds gérés macOS, le document `AWS-RunPatchBaselineWithHooks` appelle un module Python qui, à son tour, télécharge un instantané du référentiel de correctifs qui s'applique au nœud. Un sous-processus Python appelle ensuite la CLI sur le nœud afin de récupérer les informations d'installation et de mise à jour des gestionnaires de packages spécifiés, et de piloter le gestionnaire approprié pour chaque package de mise à jour.

Windows Server

Sur les nœuds Windows Server gérés, le `AWS-RunPatchBaselineWithHooks` document télécharge et invoque un PowerShell module, qui télécharge à son tour un instantané de la ligne de base des correctifs qui s'applique au nœud géré. Cet instantané de référentiel de correctifs contient une liste des correctifs approuvés qui sont compilés en interrogeant le référentiel de correctifs sur un serveur Windows Server Update Services (WSUS). Cet instantané du référentiel de correctifs est transmis à l'API Windows Update qui contrôle le téléchargement et l'installation des correctifs approuvés selon les besoins.

Chaque instantané est spécifique à un groupe de correctifs Compte AWS, à un système d'exploitation et à un identifiant de capture. L'instantané est délivré via une URL Amazon Simple Storage Service (Amazon S3) présignée, qui expire 24 heures après la création de l'instantané. Toutefois, une fois l'URL expirée, si vous souhaitez appliquer le contenu du même instantané à d'autres nœuds gérés, vous pouvez générer une nouvelle URL Amazon S3 présignée jusqu'à trois jours après la création de l'instantané. Pour ce faire, utilisez la commande get-deployable-patch-snapshot-for-instance.

Une fois que toutes les mises à jour approuvées et applicables ont été installées, et que les redémarrages nécessaires ont été effectués, des informations relatives à la conformité des correctifs sont générées sur un nœud géré et transmises à Patch Manager.

> **Note**
>
> Si le paramètre `RebootOption` est défini sur `NoReboot` dans le document `AWS-RunPatchBaselineWithHooks`, le nœud géré n'est pas redémarré après l'exécution de Patch Manager. Pour plus d'informations, consultez Nom du paramètre: `RebootOption`.

Pour plus d'informations sur l'affichage des données de conformité des correctifs, consultez A propos de la conformité des correctifs.

Étapes opérationnelles d'**`AWS-RunPatchBaselineWithHooks`**

Lorsque `AWS-RunPatchBaselineWithHooks` s'exécute, les étapes suivantes sont effectuées :

1. Analyser : une opération `Scan` utilisant `AWS-RunPatchBaseline` est exécutée sur le nœud géré, et un rapport de conformité est généré et chargé.
2. Vérifier les états des correctifs locaux : un script est exécuté pour déterminer quelles étapes seront effectuées en fonction de l'opération sélectionnée et du résultat de `Scan` à l'étape 1.
   a. Si l'opération sélectionnée est `Scan`, l'opération est marquée comme terminée. L'opération se termine.
   b. Si l'opération sélectionnée est `Install`, Patch Manager évalue le résultat de `Scan` à l'étape 1 afin de déterminer ce qu'il faut exécuter ensuite :
      i. Si aucun correctif manquant n'est détecté et qu'aucun redémarrage en attente n'est requis, l'opération passe directement à l'étape finale (étape 8), qui comprend un hook fourni par vos soins. Toutes les étapes intermédiaires sont ignorées.
      ii. Si aucun correctif manquant n'est détecté, mais qu'aucun redémarrage en attente n'est requis et que l'option de redémarrage sélectionnée est `NoReboot`, l'opération passe directement à l'étape finale (étape 8), qui comprend un hook fourni par vos soins. Toutes les étapes intermédiaires sont ignorées.
      iii. Autrement, l'opération passe à l'étape suivante.
3. Opération hook avant l'application de correctifs : le document SSM que vous avez fourni pour le premier hook de cycle de vie, `PreInstallHookDocName`, est exécuté sur le nœud géré.
4. Installer avec NoReboot : une `Install` opération avec l'option de redémarrage `NoReboot` `AWS-RunPatchBaseline` est exécutée sur le nœud géré, et un rapport de conformité est généré et téléchargé.

5. Opération hook après l'application de correctifs : le document SSM que vous avez fourni pour le premier hook de cycle de vie, `PostInstallHookDocName`, est exécuté sur le nœud géré.
6. Vérification du redémarrage : un script est exécuté afin de déterminer si un redémarrage est nécessaire pour le nœud géré et quelles sont les étapes à suivre :
   a. Si l'option de redémarrage sélectionnée est `NoReboot`, l'opération passe directement à l'étape finale (étape 8), qui comprend un hook fourni par vos soins. Toutes les étapes intermédiaires sont ignorées.
   b. Si l'option de redémarrage sélectionnée est `RebootIfNeeded`, Patch Manager vérifie si des redémarrages en attente sont requis, à partir de l'inventaire collecté à l'étape 4. Cela signifie que l'opération continue jusqu'à l'étape 7 et que le nœud géré est redémarré dans l'un des cas suivants :
      i. Patch Manager a installé un ou plusieurs correctifs. (Patch Manager n'évalue pas si un redémarrage est requis par le correctif. Le système est redémarré même si le correctif ne nécessite pas de redémarrage.)
      ii. Patch Manager détecte un ou plusieurs correctifs à l'état `INSTALLED_PENDING_REBOOT` durant l'opération Install (Installer). L'`INSTALLED_PENDING_REBOOT`état peut indiquer que l'option `NoReboot` a été sélectionnée lors de la dernière exécution de l'opération d'installation ou qu'un correctif a été installé en dehors de cette option Patch Manager depuis le dernier redémarrage du nœud géré.

      Si aucun correctif correspondant à ces critères n'est trouvé, l'opération d'application de correctifs sur un nœud géré prend fin et passe directement à l'étape finale (étape 8), qui comprend un hook fourni par vos soins. Toutes les étapes intermédiaires sont ignorées.
7. Redémarrage et rapport : une opération d'installation avec l'option de redémarrage `RebootIfNeeded` est exécutée sur le nœud géré via `AWS-RunPatchBaseline`, et un rapport de conformité est généré et chargé.
8. Opération hook après redémarrage : le document SSM que vous avez fourni pour le troisième hook de cycle de vie, `OnExitHookDocName`, est exécuté sur le nœud géré.

Pour une opération `Scan`, si l'étape 1 échoue, le processus d'exécution du document s'arrête et l'étape est signalée comme ayant échoué, bien que les étapes suivantes soient signalées comme ayant réussi.

Pour une opération `Install`, si l'une des étapes `aws:runDocument` échoue pendant l'opération, ces étapes sont signalées comme ayant échoué et l'opération passe directement à l'étape finale

(étape 8), qui comprend un hook fourni par vos soins. Toutes les étapes intermédiaires sont ignorées. Cette étape est signalée comme ayant échoué, la dernière étape indique le statut du résultat de son opération et toutes les étapes intermédiaires sont signalées comme ayant réussi.

`AWS-RunPatchBaselineWithHooks` paramètres

`AWS-RunPatchBaselineWithHooks` prend en charge six paramètres.

Le paramètre `Operation` est obligatoire.

Les paramètres `RebootOption`, `PreInstallHookDocName`, `PostInstallHookDocName` et `OnExitHookDocName` sont facultatifs.

Du point de vue technique, `Snapshot-ID` est facultatif, mais nous vous recommandons de lui donner une valeur personnalisée lorsque vous exécutez `AWS-RunPatchBaselineWithHooks` à l'extérieur d'une fenêtre de maintenance. Laissez Patch Manager fournir la valeur automatiquement lorsque le document est exécuté dans le cadre d'une opération de fenêtre de maintenance.

Paramètres

- Nom du paramètre: Operation
- Nom du paramètre: Snapshot ID
- Nom du paramètre: RebootOption
- Nom du paramètre: PreInstallHookDocName
- Nom du paramètre: PostInstallHookDocName
- Nom du paramètre: OnExitHookDocName

Nom du paramètre: **`Operation`**

Utilisation : Obligatoire.

Options : `Scan` | `Install`.

Analyser

> Lorsque vous sélectionnez l'option `Scan`, le système utilise le document `AWS-RunPatchBaseline` afin de déterminer l'état de conformité du nœud géré en matière de correctifs et transmet cette information à Patch Manager. `Scan` n'invite pas à installer les mises à

jour ou à redémarrer les nœuds gérés. Mais l'opération identifie les mises à jour manquantes qui sont approuvées et applicables au nœud.

Installation

Lorsque vous sélectionnez l'option `Install`, `AWS-RunPatchBaselineWithHooks` tente d'installer les mises à jour approuvées et applicables qu'il manque sur le nœud géré. Les informations de 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. Chaque fois qu'une mise à jour est installée sur un nœud géré, ce dernier est redémarré 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-RunPatchBaselineWithHooks`, le nœud géré n'est pas redémarré après l'exécution de Patch Manager. Pour plus d'informations, consultez Nom du paramètre: `RebootOption`).

> **Note**
>
> Si un correctif spécifié par les règles de référentiel est installé avant la mise à jour du nœud géré par Patch Manager, 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 Patch Manager pour s'assurer que les nœuds gérés d'un groupe auquel des correctifs ont été appliqués dans le cadre d'une opération individuelle disposent tous du 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-RunPatchBaselineWithHooks` 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-RunPatchBaselineWithHooks**

Mode	Bonne pratique	Détails
Exécution de AWS-RunPatchBaselineWithHooks à l'intérieur d'une fenêtre de maintenance	Ne fournissez pas d'ID d'instantané. Patch Manager le fournira pour vous.	Si vous utilisez une fenêtre de maintenance pour exécuter AWS-RunPatchBaselineWithHooks, 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-RunPatchBaselineWithHooks dans cette fenêtre de maintenance. Si vous spécifiez une valeur dans ce scénario, notez que pendant plus de trois jours, l'instantané du référentiel de correctifs peut changer. Par la suite, un nouvel instantané est généré même si vous spécifiez le même ID après l'expiration de l'instantané.
Exécution de AWS-RunPatchBaselineWithHooks à l'extérieur d'une fenêtre de maintenance	Générez et spécifiez une valeur de GUID personnalisée pour l'ID d'instantané.[1]	Si vous n'avez pas recours à une fenêtre de maintenance pour exécuter AWS-RunPatchBaselineWithHooks, nous vous recommandons de générer et de spécifier un ID d'instantané unique

Mode	Bonne pratique	Détails
		pour chaque référentiel de correctifs, en particuli er si vous exécutez le document `AWS-RunPa tchBaselineWithHoo ks` sur plusieurs nœuds gérés au cours de la même opération. Dans ce cas de figure, si vous ne spécifiez pas d'ID, Systems Manager génère un ID d'instantané différent pour chacun des nœuds gérés auxquels la commande est envoyée. Cela peut entraîner la spécifica tion d'ensembles de correctifs différents parmi les nœuds. Par exemple, si vous exécutez le document `AWS-RunPa tchBaselineWithHoo ks` directement via la fonctionnalité Run Command d'AWS Systems Manager et que vous ciblez un groupe de 50 nœuds gérés. La spécification d'un ID d'instant ané personnalisé entraîne la génération d'un instantané de référentiel unique qui permet d'évaluer et de corriger tous les nœuds gérés, garantissant ainsi un état final cohérent.

Mode	Bonne pratique	Détails

[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: **`RebootOption`**

Utilisation : Facultatif.

Options : `RebootIfNeeded` | `NoReboot`

Par défaut : `RebootIfNeeded`

> **Warning**
>
> L'option par défaut est `RebootIfNeeded`. Veillez à sélectionner l'option qui correspond à votre cas d'utilisation. Par exemple, si vos nœuds gérés doivent redémarrer immédiatement pour finaliser un processus de configuration, sélectionnez `RebootIfNeeded`. Ou, si des nœuds gérés doivent rester disponibles jusqu'à une heure de redémarrage planifiée, sélectionnez `NoReboot`.

> **Important**
>
> Nous vous déconseillons de Patch Manager les utiliser pour appliquer des correctifs à des instances de cluster dans Amazon EMR (précédemment appelé Amazon MapReduce Elastic). Ne sélectionnez pas l'option `RebootIfNeeded` pour le paramètre `RebootOption`. (Cette option est disponible dans les documents SSM Command pour l'application de correctifs sur `AWS-RunPatchBaseline`, `AWS-RunPatchBaselineAssociation` et `AWS-RunPatchBaselineWithHooks`.)
> Les commandes sous-jacentes pour l'application de correctifs à l'aide de Patch Manager utilisent les commandes `yum` et `dnf`. Par conséquent, les opérations entraînent des incompatibilités en raison de la manière dont les packages sont installés. Pour plus d'informations sur les méthodes préférées de mise à jour logicielle sur les clusters Amazon

EMR, veuillez consulter la rubrique Utilisation de l'AMI par défaut pour Amazon EMR dans le Guide de gestion Amazon EMR.

RebootIfNeeded

Lorsque vous sélectionnez l'option `RebootIfNeeded`, le nœud géré est redémarré dans l'un des cas suivants :

- Patch Manager a installé un ou plusieurs correctifs.

  Patch Manager n'évalue pas si un redémarrage est requis par le correctif. Le système est redémarré même si le correctif ne nécessite pas de redémarrage.

- Patch Manager détecte un ou plusieurs correctifs à l'état `INSTALLED_PENDING_REBOOT` durant l'opération `Install`.

  L'`INSTALLED_PENDING_REBOOT`état peut indiquer que l'option `NoReboot` a été sélectionnée lors de la dernière exécution de l'`Install`opération ou qu'un correctif a été installé en dehors de cette zone Patch Manager depuis le dernier redémarrage du nœud géré.

Dans ces deux cas, le redémarrage des nœuds gérés permet de supprimer les packages mis à jour de la mémoire, et assure la cohérence du comportement d'application des correctifs et de redémarrage sur tous les systèmes d'exploitation.

NoReboot

Lorsque vous sélectionnez l'option `NoReboot`, Patch Manager ne redémarre pas le nœud géré même s'il y a installé des correctifs pendant l'opération `Install`. Cette option est utile si vous savez qu'il n'est pas nécessaire de redémarrer vos nœuds gérés après l'application de correctifs, ou si des applications ou des processus en cours d'exécution sur un nœud ne doivent pas être perturbés par un redémarrage suite à l'application de correctifs. Elle est également utile lorsque vous souhaitez bénéficier de plus de contrôle sur le timing des redémarrages des nœuds gérés, par exemple en utilisant une fenêtre de maintenance.

> **Note**
>
> Si vous sélectionnez l'option `NoReboot` et qu'un correctif est installé, l'état du correctif est attribué au correctif `InstalledPendingReboot`. Le nœud géré, quant à lui, est marqué comme `Non-Compliant`. Après un redémarrage et l'exécution d'une opération `Scan`, l'état du nœud est mis à jour et devient `Compliant`.

Fichier de suivi de l'installation des correctifs : pour suivre l'installation des correctifs, en particulier ceux installés depuis le dernier redémarrage du système, Systems Manager gère un fichier sur le nœud géré.

> **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 correspondant au nœud géré est inexact. Dans ce cas, redémarrez le nœud et lancez une opération d'analyse des correctifs pour restaurer le fichier.

Ce fichier de suivi est stocké aux emplacements suivants sur vos nœuds gérés :

- Systèmes d'exploitation Linux :
  - `/var/log/amazon/ssm/patch-configuration/patch-states-configuration.json`
  - `/var/log/amazon/ssm/patch-configuration/patch-inventory-from-last-operation.json`
- Système d'exploitation Windows Server :
  - `C:\ProgramData\Amazon\PatchBaselineOperations\State\PatchStatesConfiguration.json`
  - `C:\ProgramData\Amazon\PatchBaselineOperations\State\PatchInventoryFromLastOperation.json`

Nom du paramètre: **`PreInstallHookDocName`**

Utilisation : Facultatif.

Par défaut : `AWS-Noop`.

La valeur à fournir pour le paramètre `PreInstallHookDocName` est le nom ou l'Amazon Resource Name (ARN) d'un document SSM de votre choix. Vous pouvez fournir le nom d'un document AWS géré ou le nom ou l'ARN d'un document SSM personnalisé que vous avez créé ou qui a été partagé avec vous. (Pour un document SSM qui a été partagé avec vous à partir d'un autre document Compte AWS, vous devez spécifier l'ARN complet de la ressource, par exemple`arn:aws:ssm:us-east-2:123456789012:document/MySharedDocument`.)

Le document SSM spécifié par vos soins est exécuté avant l'opération `Install` et effectue toutes les actions prises en charge par SSM Agent, comme l'exécution d'un script shell pour assurer la surveillance de l'état des applications avant l'installation des correctifs sur le nœud géré. (Pour obtenir la liste des actions, consultez Référence de plug-in de document Command). Par défaut, le document SSM porte le nom `AWS-Noop`. Celui-ci n'effectue aucune opération sur le nœud géré.

Pour de plus amples informations sur la création d'un document SSM personnalisé, veuillez consulter Création du contenu du document SSM.

Nom du paramètre: **`PostInstallHookDocName`**

Utilisation : Facultatif.

Par défaut : `AWS-Noop`.

La valeur à fournir pour le paramètre `PostInstallHookDocName` est le nom ou l'Amazon Resource Name (ARN) d'un document SSM de votre choix. Vous pouvez fournir le nom d'un document AWS géré ou le nom ou l'ARN d'un document SSM personnalisé que vous avez créé ou qui a été partagé avec vous. (Pour un document SSM qui a été partagé avec vous à partir d'un autre document Compte AWS, vous devez spécifier l'ARN complet de la ressource, par exemple`arn:aws:ssm:us-east-2:123456789012:document/MySharedDocument`.)

Le document SSM que vous spécifiez est exécuté après l'opération `Install with NoReboot` et effectue toutes les actions prises en charge par SSM Agent, par exemple un script shell pour installer des mises à jour tierces avant le redémarrage. (Pour obtenir la liste des actions, consultez Référence de plug-in de document Command). Par défaut, le document SSM porte le nom `AWS-Noop`. Celui-ci n'effectue aucune opération sur le nœud géré.

Pour de plus amples informations sur la création d'un document SSM personnalisé, veuillez consulter Création du contenu du document SSM.

Nom du paramètre: **`OnExitHookDocName`**

Utilisation : Facultatif.

Par défaut : `AWS-Noop`.

La valeur à fournir pour le paramètre `OnExitHookDocName` est le nom ou l'Amazon Resource Name (ARN) d'un document SSM de votre choix. Vous pouvez fournir le nom d'un document AWS géré ou le nom ou l'ARN d'un document SSM personnalisé que vous avez créé ou qui a été partagé avec

vous. (Pour un document SSM partagé avec vous à partir d'un Compte AWS différent, vous devez spécifier l'ARN complet de la ressource, `arn:aws:ssm:us-east-2:123456789012:document/MySharedDocument` par exemple.)

Le document SSM spécifié par vos soins est exécuté après l'opération de redémarrage du nœud géré et effectue toutes les actions prises en charge par SSM Agent, comme l'exécution d'un script shell pour vérifier l'état du nœud une fois l'opération d'application des correctifs terminée. (Pour obtenir la liste des actions, consultez Référence de plug-in de document Command). Par défaut, le document SSM porte le nom `AWS-Noop`. Celui-ci n'effectue aucune opération sur le nœud géré.

Pour de plus amples informations sur la création d'un document SSM personnalisé, veuillez consulter Création du contenu du document SSM.

Exemple de scénario pour l'utilisation du paramètre InstallOverrideList dans **`AWS-RunPatchBaseline`** ou **`AWS-RunPatchBaselineAssociation`**

Vous pouvez utiliser le paramètre `InstallOverrideList` lorsque vous souhaitez remplacer les correctifs spécifiés par le référentiel des correctifs par défaut actuel dans Patch Manager, une des fonctionnalités de AWS Systems Manager. Cette rubrique fournit des exemples sur l'utilisation de ce paramètre pour atteindre les objectifs suivants :

- Appliquez différents ensembles de correctifs à un groupe cible de nœuds gérés.
- 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 nœuds gérés 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 le même référentiel 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 fenêtre de maintenance qui exécute `AWS-RunPatchBaseline` ou `AWS-RunPatchBaselineAssociation` 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 Simple Storage Service (Amazon S3).
4. Créez une deuxième fenêtre de maintenance qui s'exécute une fois par mois. Toutefois, pour la tâche Run Command que vous inscrivez pour cette fenêtre de maintenance, spécifiez l'emplacement de votre liste de remplacement.

Résultat : seuls les correctifs `Security`, tels que définis dans votre ligne de base 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.

Pour plus d'informations et de listes d'exemples, veuillez consulter Nom du paramètre: `InstallOverrideList`.

Utilisation du BaselineOverride paramètre

Vous pouvez définir les préférences d'application des correctifs lors de l'exécution à l'aide de la fonction de remplacement de la ligne de base dansPatch Manager, une fonctionnalité de. AWS Systems Manager Pour cela, spécifiez un compartiment Amazon Simple Storage Service (Amazon S3) contenant un objet JSON avec une liste de référentiels de correctifs. L'opération d'application de correctifs utilise les référentiels fournis dans l'objet JSON et correspondant au système d'exploitation hôte au lieu d'appliquer les règles du référentiel de correctifs par défaut.

> **Note**
>
> Sauf lorsqu'une opération d'application de correctifs utilise une politique de correctifs, l'utilisation du `BaselineOverride` paramètre n'annule pas la conformité aux correctifs de la ligne de base fournie dans le paramètre. Les résultats de sortie sont enregistrés dans les journaux Stdout à partir deRun Command, une fonctionnalité de. AWS Systems Manager Les résultats n'impriment que les packages marqués comme `NON_COMPLIANT`.

Cela signifie que le package est marqué comme `Missing`, `Failed`, `InstalledRejected` ou `InstalledPendingReboot`.
Toutefois, lorsqu'une opération de correctif utilise une politique de correctif, le système transmet le paramètre de remplacement du compartiment S3 associé et la valeur de conformité est mise à jour pour le nœud géré. Pour plus d'informations sur les comportements liés aux politiques relatives aux correctifs, consultezUtilisation des stratégies de correctifs Quick Setup.

Utilisation du remplacement du référentiel de correctifs par les paramètres d'ID d'instantané ou de liste de remplacement d'installation

Dans deux cas précis, le remplacement du référentiel de correctifs a un comportement remarquable.

Utilisation simultanée du remplacement du référentiel et de l'ID d'instantané

Les ID d'instantané permettent d'appliquer les mêmes correctifs sur tous les nœuds gérés associés à une commande particulière. Par exemple, si vous appliquez des correctifs sur 1 000 nœuds à la fois, il s'agit des mêmes correctifs.

Lorsque vous utilisez à la fois un ID d'instantané et un remplacement du référentiel de correctifs, l'ID d'instantané a priorité sur le remplacement du référentiel de correctifs. Les règles de remplacement du référentiel seront utilisées, mais elles ne seront évaluées qu'une seule fois. Dans l'exemple précédent, les correctifs appliqués à vos 1 000 nœuds gérés seront toujours les mêmes. Si, à mi-parcours de l'opération d'application de correctifs, vous avez modifié le fichier JSON dans le compartiment S3 référencé pour en faire quelque chose de différent, les correctifs appliqués ne changeront pas. Cela vient du fait que l'ID d'instantané a été fourni.

Utilisation simultanée du remplacement du référentiel et de la liste de remplacement d'installation

Vous ne pouvez pas utiliser ces deux paramètres en même temps. Le document d'application de correctifs échoue si les deux paramètres sont fournis, et il ne procède alors à aucune analyse ou installation sur le nœud géré.

Exemples de code

L'exemple de code suivant pour Python montre la génération du remplacement du référentiel de correctifs.

```
import boto3
```

```
import json

ssm = boto3.client('ssm')
s3 = boto3.resource('s3')
s3_bucket_name = 'my-baseline-override-bucket'
s3_file_name = 'MyBaselineOverride.json'
baseline_ids_to_export = ['pb-0000000000000000', 'pb-0000000000000001']

baseline_overrides = []
for baseline_id in baseline_ids_to_export:
    baseline_overrides.append(ssm.get_patch_baseline(
        BaselineId=baseline_id
    ))

json_content = json.dumps(baseline_overrides, indent=4, sort_keys=True, default=str)
s3.Object(bucket_name=s3_bucket_name, key=s3_file_name).put(Body=json_content)
```

Voici comment se produit un remplacement du référentiel de correctifs :

```
[
    {
        "ApprovalRules": {
            "PatchRules": [
                {
                    "ApproveAfterDays": 0,
                    "ComplianceLevel": "UNSPECIFIED",
                    "EnableNonSecurity": false,
                    "PatchFilterGroup": {
                        "PatchFilters": [
                            {
                                "Key": "PRODUCT",
                                "Values": [
                                    "*"
                                ]
                            },
                            {
                                "Key": "CLASSIFICATION",
                                "Values": [
                                    "*"
                                ]
                            },
                            {
                                "Key": "SEVERITY",
```

```
                            "Values": [
                                "*"
                            ]
                        }
                    ]
                }
            }
        ]
    },
    "ApprovedPatches": [],
    "ApprovedPatchesComplianceLevel": "UNSPECIFIED",
    "ApprovedPatchesEnableNonSecurity": false,
    "GlobalFilters": {
        "PatchFilters": []
    },
    "OperatingSystem": "AMAZON_LINUX_2",
    "RejectedPatches": [],
    "RejectedPatchesAction": "ALLOW_AS_DEPENDENCY",
    "Sources": []
},
{
    "ApprovalRules": {
        "PatchRules": [
            {
                "ApproveUntilDate": "2021-01-06",
                "ComplianceLevel": "UNSPECIFIED",
                "EnableNonSecurity": true,
                "PatchFilterGroup": {
                    "PatchFilters": [
                        {
                            "Key": "PRODUCT",
                            "Values": [
                                "*"
                            ]
                        },
                        {
                            "Key": "CLASSIFICATION",
                            "Values": [
                                "*"
                            ]
                        },
                        {
                            "Key": "SEVERITY",
                            "Values": [
```

```
                                    "*"
                                ]
                            }
                        ]
                    }
                }
            ]
        },
        "ApprovedPatches": [
            "open-ssl*"
        ],
        "ApprovedPatchesComplianceLevel": "UNSPECIFIED",
        "ApprovedPatchesEnableNonSecurity": false,
        "GlobalFilters": {
            "PatchFilters": []
        },
        "OperatingSystem": "CENTOS",
        "RejectedPatches": [
            "python*"
        ],
        "RejectedPatchesAction": "ALLOW_AS_DEPENDENCY",
        "Sources": []
    }
]
```

À propos des références de correctifs

Les rubriques de cette section fournissent des informations sur le fonctionnement des référentiels de correctifs dans la fonctionnalité Patch Manager d'AWS Systems Manager lorsque vous exécutez une opération `Scan` ou `Install` sur vos nœuds gérés.

Rubriques

- À propos des références de correctifs prédéfinies et personnalisées
- À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés
- À propos des groupes de correctifs
- À propos de la correction d'applications publiées par Microsoft sur Windows Server

À propos des références de correctifs prédéfinies et personnalisées

Patch Manager, une fonctionnalité de AWS Systems Manager, fournit des lignes de base de correctifs prédéfinies pour chacun des systèmes d'exploitation pris en charge parPatch Manager. 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 lignes de base 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 plus d'informations, consultez À propos des références personnalisées et Utilisation des référentiels de correctifs personnalisés.

> **Note**
>
> Les informations de cette rubrique s'appliquent, quels que soient la méthode ou le type de configuration que vous utilisez pour vos opérations d'application de correctifs :
>
> - Une politique de correctifs configurée dans Quick Setup
> - Une option de gestion des hôtes configurée dans Quick Setup
> - Une fenêtre de maintenance pour exécuter un correctif `Scan` ou une tâche `Install`
> - Une opération Patch now (Appliquer les correctifs maintenant) à la demande

Rubriques

À propos des références prédéfinies

Le tableau ci-dessous décrit les références de correctifs prédéfinies fournies avec Patch Manager.

Pour plus d'informations sur les versions de chaque système d'exploitation que Patch Manager prend en charge, consultez Conditions préalables requises Patch Manager.

Nom	Système d'exploitation pris en charge	Détails
`AWS-AlmaLinuxDefaultPatchBaseline`	AlmaLinux	Approuve tous les correctifs de système d'exploitation qui sont classés dans la section « Security (Sécurité) » et qui ont un niveau de sévérité « Critical (Critique) » ou « Important ». Approuve également tous les correctifs classés comme « Bugfix » (Correctif de bogue). L'approbation automatique des correctifs se fait 7 jours après leur publication ou leur mise à jour.[1]
`AWS-AmazonLinuxDefaultPatchBaseline`	Amazon Linux 1	Approuve tous les correctifs de système d'exploitation qui sont classés dans la section « Security (Sécurité) » et qui ont un niveau de sévérité « Critical (Critique) » ou « Important ». Avec une classification « Bugfix », l'approbation automatique de tous les correctifs est possible. L'approbation automatique des correctifs se fait 7 jours après leur publication ou leur mise à jour.[1]
`AWS-AmazonLinux2DefaultPatchBaseline`	Amazon Linux 2	Approuve tous les correctifs de système d'exploitation qui sont classés dans la section « Security (Sécurité

Nom	Système d'exploitation pris en charge	Détails
		) » et qui ont un niveau de sévérité « Critical (Critique) » ou « Important ». Avec une classification « Bugfix », l'approbation automatique de tous les correctifs est possible. L'approbation automatique des correctifs se fait 7 jours après leur publication.[1]
`AWS-AmazonLinux2022DefaultPatchBaseline`	Amazon Linux 2022	Approuve tous les correctifs de système d'exploitation qui sont classés dans la section « Security (Sécurité) » et qui ont un niveau de sévérité « 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.

Nom	Système d'exploitation pris en charge	Détails
`AWS-AmazonLinux2023DefaultPatchBaseline`	Amazon Linux 2023	Approuve tous les correctifs de système d'exploitation qui sont classés dans la section « Security (Sécurité) » et qui ont un niveau de sévérité « 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 et CentOS Stream	Approuvez toutes les mises à jour 7 jours après leur disponibilité, notamment 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 les dates de publication fiables sont indisponibles dans le référentiel.

Nom	Système d'exploitation pris en charge	Détails
`AWS-MacOSDefaultPatchBaseline`	macOS	Approuve tous les correctifs de système d'exploitation classifiés comme « liés à la sécurité » Approuve également tous les packages faisant l'objet d'une mise à jour.
`AWS-OracleLinuxDefaultPatchBaseline`	Oracle Linux	Approuve tous les correctifs de système d'exploitation qui sont classés dans la section « Security (Sécurité) » et qui ont un niveau de sévérité « Important » ou « Moderate (Modéré) ». Approuve également tous les correctifs classés comme « Bugfix » (Correctif de bogue) 7 jours après leur publication. L'approbation automatique des correctifs se fait 7 jours après leur publication ou leur mise à jour.[1]

Nom	Système d'exploitation pris en charge	Détails
`AWS-DefaultRaspbianPatchBaseline`	Raspberry Pi OS	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 les dates de publication fiables sont indisponibles dans le référentiel.
`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 sévérité « Critical (Critique) » ou « Important ». Approuve également tous les correctifs classés comme « Bugfix » (Correctif de bogue). L'approbation automatique des correctifs se fait 7 jours après leur publication ou leur mise à jour.[1]

Nom	Système d'exploitation pris en charge	Détails
`AWS-RockyLinuxDefaultPatchBaseline`	Rocky Linux	Approuve tous les correctifs de système d'exploitation qui sont classés dans la section « Security (Sécurité) » et qui ont un niveau de sévérité « Critical (Critique) » ou « Important ». Approuve également tous les correctifs classés comme « Bugfix » (Correctif de bogue). L'approbation automatique des correctifs se fait 7 jours après leur publication ou leur mise à jour.[1]
`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 sévérité « Critical (Critique) » ou « Important ». L'approbation automatique des correctifs se fait 7 jours après leur publication ou leur mise à jour.[1]

Nom	Système d'exploitation pris en charge	Détails
`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 les dates de publication fiables sont indisponibles dans le référentiel.
`AWS-DefaultPatchBaseline`	Windows Server	Approuve tous les correctifs du système Windows Server d'exploitation classés comme « CriticalUpdates » ou « SecurityUpdates » et dont le niveau de gravité MSRC est « Critique » ou « Important ». L'approbation automatique des correctifs se fait 7 jours après leur publication ou leur mise à jour.[2]

Nom	Système d'exploitation pris en charge	Détails
`AWS-WindowsPredefinedPatchBaseline-OS`	Windows Server	Approuve tous les correctifs du système Windows Server d'exploitation classés comme « CriticalUpdates » ou « SecurityUpdates » et dont le niveau de gravité MSRC est « Critique » ou « Important ». L'approbation automatique des correctifs se fait 7 jours après leur publication ou leur mise à jour.[2]
`AWS-WindowsPredefinedPatchBaseline-OS-Applications`	Windows Server	Pour le système Windows Server d'exploitation, approuve tous les correctifs classés comme « » ou CriticalUpdates « SecurityUpdates » et dont le niveau de gravité MSRC est « Critique » ou « Important ». Pour les applications publiées par Microsoft, approuve tous les correctifs. Les correctifs de système d'exploitation et d'applications sont automatiquement approuvés 7 jours après leur publication.[2]

[1] Pour Amazon Linux 1 et Amazon Linux 2, le délai d'attente de 7 jours avant l'approbation automatique des correctifs est calculé à partir d'une `Updated Date` valeur en`updateinfo.xml`, et non d'une `Release Date` valeur. Divers facteurs peuvent affecter la valeur `Updated Date`. Les autres systèmes d'exploitation gèrent les dates de sortie ainsi que de mise à jour de façon différente.

Pour des informations vous permettant d'éviter des résultats inattendus avec les délais d'approbation automatique, consultez Calcul des dates de sortie et des mises à jour des packages.

[2] Pour Windows Server, les références par défaut incluent un délai d'approbation automatique de 7 jours. Pour installer un correctif dans les 7 jours suivant sa publication, vous devez créer une base de référence personnalisée.

À propos des références personnalisées

Si vous créez votre propre référentiel de correctifs, vous pouvez choisir les correctifs à approuver automatiquement en utilisant les catégories suivantes.

- Système d'exploitation : Windows Server, Amazon Linux, Ubuntu Server, etc.
- 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 publiées par Microsoft Windows Server uniquement) : par exemple, Word 2016, BizTalk Server, etc.
- Classification : par exemple, mises à jour critiques, mises à jour de sécurité, etc.
- Sévérité : 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.

Le délai d'approbation automatique correspond au nombre de jours à attendre après la publication ou la dernière mise à jour du correctif, ceci avant que ce dernier ne soit automatiquement approuvé pour application. Par exemple, si vous créez une règle à l'aide de la classification `CriticalUpdates` et que vous la configurez pour un délai d'approbation automatique de 7 jours, un nouveau correctif critique publié le 7 juillet sera automatiquement approuvé le 14 juillet.

> **Note**
>
> Si un référentiel Linux ne fournit pas d'informations sur la date de sortie des packages, Systems Manager utilise l'heure de création du package comme délai d'approbation automatique pour Amazon Linux 1, Amazon Linux 2 et RHEL CentOS. Si le système ne peut pas trouver le temps de génération du package, Systems Manager traite le délai d'approbation automatique comme ayant une valeur de zéro.

Lorsque vous indiquez une date limite d'auto-approbation, Patch Manager tous les correctifs publiés ou mis à jour à cette date ou avant sont automatiquement appliqués. Par exemple, si vous indiquez le 7 juillet 2023 comme date limite, aucun correctif publié ou mis à jour le 8 juillet 2023 ou après ne sera installé automatiquement.

> **Note**
>
> Lorsque vous créez un référentiel de correctifs personnalisé, vous pouvez spécifier un niveau de sévérité de conformité pour les correctifs approuvés par ce référentiel de correctifs, tel que `Critical` ou `High`. Si l'état des correctifs d'un correctif approuvé est indiqué `Missing`, le niveau de sévérité de conformité global indiqué pour le référentiel de correctifs est le niveau de sévérité que vous avez spécifié.

Gardez les points suivants à l'esprit lorsque vous créez un référentiel de correctifs :

- Patch Manager fournit un référentiel de correctifs prédéfini 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 référentiel de correctifs et que vous la définissez comme le référentiel par défaut pour le type de système d'exploitation correspondant.

  > **Note**
  >
  > Pour Windows Server, trois référentiels de correctifs prédéfinis sont fournis. Les référentiels de correctifs `AWS-DefaultPatchBaseline` et `AWS-WindowsPredefinedPatchBaseline-OS` prennent uniquement en charge les mises à jour du système d'exploitation Windows. `AWS-DefaultPatchBaseline` est utilisé comme référentiel de correctifs par défaut pour les nœuds gérés Windows Server, sauf si vous en

spécifiez un autre. Ces deux référentiels de correctifs ont des paramètres de configuration identiques. Le plus récent des deux, `AWS-WindowsPredefinedPatchBaseline-OS`, a été créé pour le différencier du troisième référentiel de correctifs prédéfini pour Windows Server. Ce référentiel de correctifs, `AWS-WindowsPredefinedPatchBaseline-OS-Applications`, peut être utilisé pour appliquer des correctifs à la fois au système d'exploitation Windows Server et aux applications prises en charge publiées par Microsoft.

- Pour les serveurs et les machines virtuelles (VM) sur site, Patch Manager tente d'utiliser votre référentiel de correctifs personnalisé par défaut. S'il n'existe aucun référentiel de correctifs personnalisée par défaut, le système utilise le référentiel de correctifs prédéfinie pour le système d'exploitation correspondant.
- Si un correctif est répertorié comme approuvé et refusé dans la même référentiel de correctifs, le correctif est refusé.
- Vous ne pouvez définir qu'un seul référentiel de correctifs par nœud géré.
- Les formats de noms de package que vous pouvez ajouter à la liste des correctifs approuvés et rejetés pour un référentiel de correctifs dépendent du type de système d'exploitation auquel vous appliquez des correctifs.

  Pour obtenir des informations sur les formats acceptés de 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.
- Si vous utilisez une configuration de politique de correctifs dans Quick Setup, les mises à jour que vous apportez aux référentiels de correctifs personnalisés sont synchronisées avec Quick Setup une fois par heure.

  Si un référentiel de correctifs personnalisé référencé dans une politique de correctifs est supprimé, une bannière s'affiche sur la page Quick Setup Configuration details (Détails de configuration) de votre politique de correctifs. La bannière vous informe que la politique de correctifs fait référence à un référentiel de correctifs qui n'existe plus et que les opérations d'application de correctifs suivantes échoueront. Dans ce cas, revenez à la page Quick Setup Configurations, sélectionnez la configuration Patch Manager, puis choisissez Actions, Edit configuration (Modifier la configuration). Le nom du référentiel de correctifs supprimé est surligné et vous devez sélectionner un nouveau référentiel de correctifs pour le système d'exploitation concerné.

Pour plus d'informations sur la création d'un référentiel de correctifs, consultez Utilisation des référentiels de correctifs personnalisés et Didacticiel : application de correctifs à un environnement de serveur (AWS CLI).

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 référentiel 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 1, Amazon Linux 2, Amazon Linux 2022, Amazon Linux 2023, CentOS et Red Hat Enterprise Linux () Oracle LinuxRHEL
- Debian Server, Raspberry Pi OS (anciennement Raspbian) et Ubuntu Server
- SUSE Linux Enterprise Server (SLES)

Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022, Amazon Linux 2023, CentOS et Red Hat Enterprise Linux () Oracle LinuxRHEL

Gestionnaire de packages : YUM, à l'exception d'Amazon Linux 2022, d'Amazon Linux 2023, de RHEL 8 et de CentOS 8, qui utilisent DNF comme gestionnaire de packages

Correctifs approuvés : pour les correctifs approuvés, vous pouvez spécifier l'un des éléments suivants :

- ID Bugzilla, au format `1234567` (Le système traite les chaînes composées uniquement de chiffres en tant qu'ID Bugzilla.)
- ID CVE, au format `CVE-2018-1234567`
- ID Advisory, dans des formats tels que `RHSA-2017:0864` et `ALAS-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, Raspberry Pi OS (anciennement Raspbian) et Ubuntu Server

Gestionnaire de package : APT

Correctifs approuvés et correctifs rejetés : pour les correctifs approuvés comme rejetés, spécifiez l'un des éléments suivants :

- Noms de package, au format `ExamplePkg33`

> **Note**
>
> Pour les listes Debian Server, Raspberry Pi OS et Ubuntu Server, n'incluez pas d'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 correctifs rejetés : pour les listes de correctifs approuvés comme 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 macOS

Gestionnaires de packages pris en charge : softwareupdate, programme d'installation, Brew, Brew Cask

Correctifs approuvés et correctifs rejetés : pour les listes de correctifs approuvés et de correctifs rejetés, vous pouvez spécifier des noms de packages complets, dans l'un des formats suivants :

- `XProtectPlistConfigData`
- `MRTConfigData`

Les caractères génériques ne sont pas pris en charge dans les listes de correctifs approuvés et rejetés pour macOS.

Formats de noms de package pour les systèmes d'exploitation Windows

Pour les systèmes d'exploitation Windows, spécifiez des correctifs à l'aide des ID de la base de connaissance Microsoft et des ID de bulletins de sécurité Microsoft ; par exemple :

```
KB2032276,KB2124261,MS10-048
```

À propos des groupes de correctifs

> **Important**
>
> Les groupes de correctifs ne sont pas utilisés dans les opérations d'application de correctifs basées sur des politiques de correctifs. Pour obtenir des informations sur l'utilisation des politiques de correctifs, consultez la rubrique Utilisation des stratégies de correctifs Quick Setup.

Vous pouvez utiliser un groupe de correctifs pour associer des nœuds gérés à une ligne de base de correctifs spécifique dansPatch Manager, une fonctionnalité de AWS Systems Manager. Les groupes de correctifs vous permettent de vous assurer que vous déployez les correctifs appropriés, conformément aux règles de référentiel de correctifs associées, pour le groupe de nœuds 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 un référentiel de correctifs appropriée.

Lorsque vous exécutez `AWS-RunPatchBaseline`, vous pouvez cibler les nœuds gérés à l'aide de leur ID ou de leurs balises. SSM Agent et Patch Manager déterminent alors quel est le référentiel de correctifs à utiliser en fonction de la valeur de groupe de correctifs que vous avez ajoutée au nœud géré.

Vous créez un groupe de correctifs en utilisant des balises Amazon Elastic Compute Cloud (Amazon EC2). Contrairement à d'autres scénarios de balisage dans Systems Manager, un groupe de correctifs doit être défini avec la clé de balise `Patch Group` ou `PatchGroup`. La clé est sensible à la casse. Vous pouvez spécifier n'importe quelle valeur pour vous aider à identifier et à cibler les ressources de ce groupe, par exemple « serveurs web » ou « US-EAST-PROD », cependant la clé doit être `Patch Group` ou `PatchGroup`.

Après avoir créé un groupe de correctifs et attribué des balises aux nœuds gérés, vous pouvez enregistrer le groupe de correctifs auprès d'un référentiel de correctifs. L'enregistrement du groupe de correctifs auprès d'un référentiel de correctifs garantit que les nœuds du groupe de correctifs utiliseront les règles définies dans le référentiel de correctifs associé.

Pour de plus amples informations sur la création d'un groupe de correctifs et son association à un référentiel de correctifs, consultez Utilisation des groupes de correctifs et Ajout d'un groupe de correctifs à un référentiel de correctifs.

Pour voir un exemple de création d'un référentiel de correctifs et de groupes de correctifs à l'aide de l' AWS Command Line Interface (AWS CLI), consultez Didacticiel : application de correctifs à un environnement de serveur (AWS CLI). Pour plus d'informations sur les balises Amazon EC2, consultez la section Marquer vos ressources Amazon EC2 dans le guide de l'utilisateur Amazon EC2.

Comment ça marche

Lorsque le système applique un référentiel de correctifs à un nœud géré, SSM Agent vérifie si une valeur de groupe de correctifs est définie pour le nœud. Si le nœud est attribué à un groupe de correctifs, Patch Manager vérifie alors quel référentiel de correctifs est enregistré pour ce groupe. S'il existe un référentiel de correctifs pour ce groupe, Patch Manager demande à l'SSM Agent de l'utiliser. Si un nœud n'est attribué à aucun groupe de correctifs, Patch Manager demande automatiquement à SSM Agent d'utiliser le référentiel de correctifs par défaut.

> **Important**
>
> Un nœud géré ne peut appartenir qu'à un seul groupe de correctifs.
> Un groupe de correctifs peut être enregistré avec un seule référentiel de correctifs pour chaque type de système d'exploitation.
> Vous ne pouvez pas appliquer la `Patch Group` balise (avec un espace) à une instance Amazon EC2 si l'option Autoriser les balises dans les métadonnées d'instance est activée sur celle-ci. L'autorisation des identifications dans les métadonnées d'instance empêche les noms de clés d'identification de contenir des espaces. Si vous avez autorisé les balises dans les métadonnées des instances EC2, vous devez utiliser la clé de la balise `PatchGroup` (sans espace).

Le schéma suivant illustre un exemple général des processus exécutés par Systems Manager lors de l'envoi à votre parc de serveurs d'une tâche Run Command d'application de correctifs à l'aide du Patch Manager. Un processus similaire est utilisé lorsqu'une fenêtre de maintenance est configurée pour envoyer une commande d'application de correctifs à l'aide du Patch Manager.

Cet exemple illustre trois groupes d'instances EC2 pour Windows Server avec les balises suivantes appliquées :

Groupe d'instances EC2	Balises
Groupe 1	`key=OS,value=Windows`

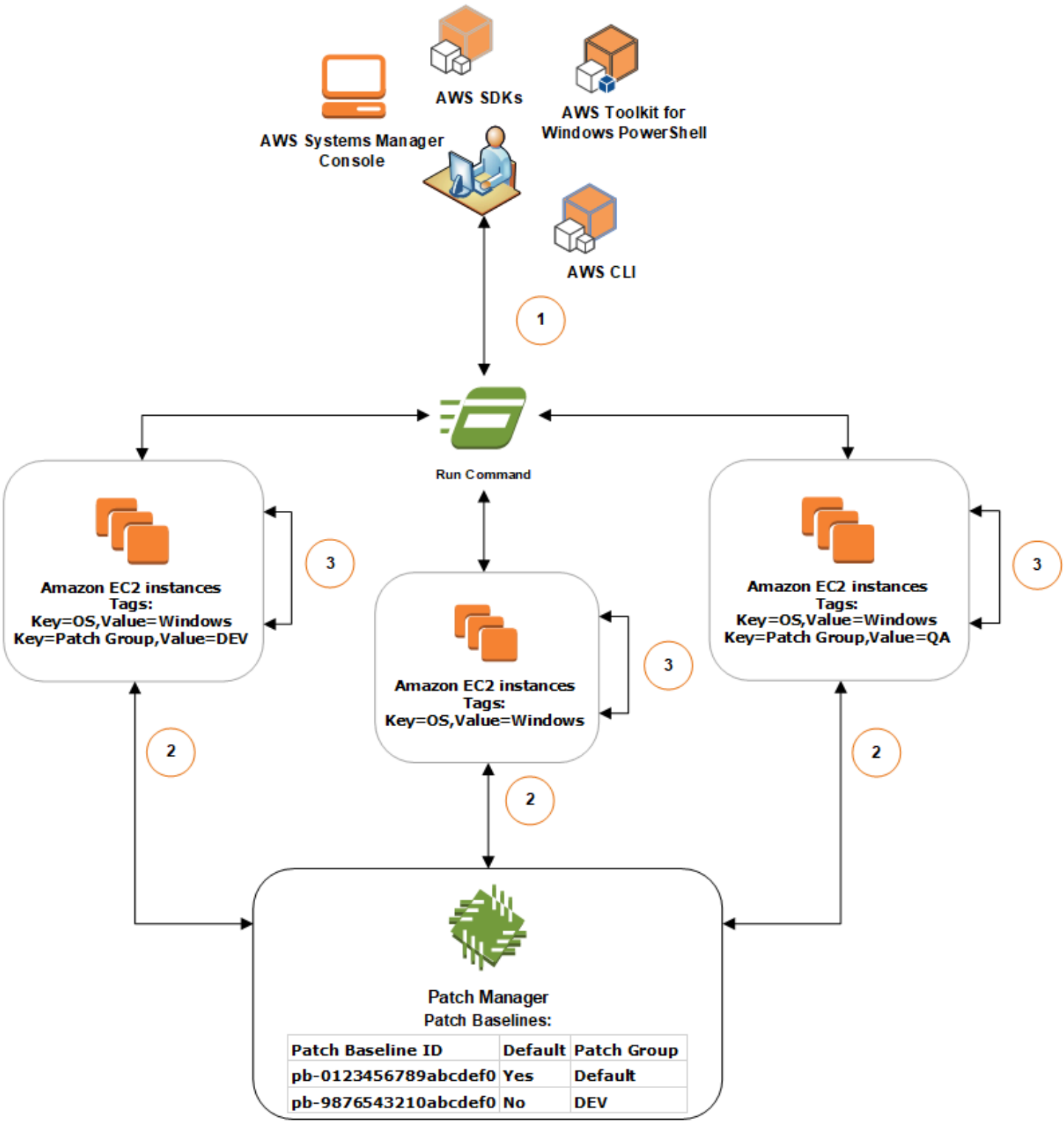
Groupe d'instances EC2	Balises
	`key=PatchGroup,value=DEV`
Groupe 2	`key=OS,value=Windows`
Groupe 3	`key=OS,value=Windows` `key=PatchGroup,value=QA`

Dans cet exemple, nous avons également ces deux référentiels de correctifs Windows Server :

ID de référence de correctif	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 schéma suivant montre comment Patch Manager détermine les référentiel de correctifs utiliser dans les opérations d'application de correctifs.



La procédure générale d'analyse ou d'installation des correctifs à l'aide de Run Command, une des fonctionnalités de AWS Systems Manager et de Patch Manager se présente comme suit :

1. Envoyer une commande au correctif : utilisez la console Systems Manager, le SDK AWS Command Line Interface (AWS CLI) ou AWS Tools for Windows PowerShell pour envoyer une Run Command tâche à l'aide du document`AWS-RunPatchBaseline`. Le diagramme illustre une tâche Run Command d'application de correctifs aux instances gérées en ciblant la balise `key=OS,value=Windows`.
2. Détermination du référentiel de correctifs : SSM Agent vérifie les balises de groupe de correctifs appliquées à l'instance EC2 et interroge Patch Manager concernant le référentiel de correctifs correspondante.
   - Mise en correspondance de la valeur du groupe de correctifs associée avec le référentiel de correctifs:
     1. L'SSM Agent, qui est installé sur les instances EC2 dans un groupe, reçoit la commande émise à l'étape 1 lui indiquant de lancer une opération d'application de correctifs. L'SSM Agent vérifie que la valeur-balise de groupe de correctifs appliquée aux instances EC2 est `DEV` et interroge Patch Manager concernant le référentiel de correctifs associée.
     2. Patch Manager vérifie que le référentiel de correctifs `pb-9876543210abcdef0` est associée au groupe de correctifs `DEV` et informe l'SSM Agent.
     3. L'SSM Agent récupère un instantané de référentiel de correctifs depuis Patch Manager en fonction des règles d'approbation et des exceptions configurées dans `pb-9876543210abcdef0`, puis passe à l'étape suivante.
   - Aucune balise de groupe de correctifs n'est ajoutée à l'instance :
     1. SSM Agent, qui est installé sur les instances EC2 du groupe deux, reçoit la commande émise à l'étape 1 pour commencer une opération de correction. SSM Agent valide que les instances EC2 n'ont pas de balise `Patch Group` ou `PatchGroup` appliquée et, par conséquent, SSM Agent demande Patch Manager la ligne de base de correction Windows par défaut.
     2. Patch Manager vérifie que le référentiel de correctifs Windows Server par défaut est `pb-0123456789abcdef0` et en avertit l'SSM Agent.
     3. L'SSM Agent récupère un instantané de référentiel de correctifs depuis Patch Manager en fonction des règles d'approbation et des exceptions configurées dans le référentiel de correctifs par défaut `pb-0123456789abcdef0`, puis passe à l'étape suivante.
   - Aucune valeur de groupe de correctifs correspondante n'est associée à un référentiel de correctifs:

1. L'SSM Agent, qui est installé sur les instances EC2 du groupe trois, reçoit la commande émise à l'étape 1 lui indiquant de lancer une opération d'application de correctifs. L'SSM Agent vérifie que la valeur-balise de groupe de correctifs appliquée aux instances EC2 est `QA` et interroge Patch Manager concernant le référentiel de correctifs associée.
2. Patch Manager ne trouve aucun référentiel de correctifs associée au groupe `QA`.
3. Patch Manager demande à l'SSM Agent d'utiliser le référentiel de correctifs Windows par défaut `pb-0123456789abcdef0`.
4. L'SSM Agent récupère un instantané de référentiel de correctifs depuis Patch Manager en fonction des règles d'approbation et des exceptions configurées dans le référentiel de correctifs par défaut `pb-0123456789abcdef0`, puis passe à l'étape suivante.

3. Recherche ou installation des correctifs : Après avoir déterminé le référentiel de correctifs appropriée à utiliser, l'SSM Agent 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 référentiel de correctifs fourni par Patch Manager.

Plus d'informations

- Comprendre les valeurs d'état de conformité des correctifs

À propos de la correction d'applications publiées par Microsoft sur Windows Server

Utilisez les informations de cette rubrique pour vous préparer à corriger des applications sur Windows Server en utilisant Patch Manager, une fonctionnalité de AWS Systems Manager.

Correction d'applications Microsoft

La prise en charge de l'installation de correctifs destinés aux applications sur les nœuds gérés par Windows Server s'applique uniquement aux applications publiées par Microsoft.

> **Note**
>
> Dans certains cas, Microsoft publie des correctifs pour les applications qui ne spécifient pas de date et d'heure de mise à jour. La date et l'heure `01/01/1970` sont alors fournies par défaut.

Référentiels de correctifs pour corriger des applications publiées par Microsoft

Pour Windows Server, trois référentiels de correctifs prédéfinis sont fournis. Les référentiels de correctifs `AWS-DefaultPatchBaseline` et `AWS-WindowsPredefinedPatchBaseline-OS` prennent uniquement en charge les mises à jour du système d'exploitation Windows. `AWS-DefaultPatchBaseline` est utilisé comme référentiel de correctifs par défaut pour les nœuds gérés Windows Server, sauf si vous en spécifiez un autre. Ces deux référentiels de correctifs ont des paramètres de configuration identiques. Le plus récent des deux, `AWS-WindowsPredefinedPatchBaseline-OS`, a été créé pour le différencier du troisième référentiel de correctifs prédéfini pour Windows Server. Ce référentiel de correctifs, `AWS-WindowsPredefinedPatchBaseline-OS-Applications`, peut être utilisé pour appliquer des correctifs à la fois au système d'exploitation Windows Server et aux applications prises en charge publiées par Microsoft.

Vous pouvez également créer un référentiel de correctifs personnalisé pour mettre à jour les applications publiées par Microsoft sur les machines Windows Server.

Prise en charge de l'application des correctifs publiés par Microsoft sur les serveurs sur site, les périphériques, les machines virtuelles et d'autres nœuds non EC2

Pour appliquer des correctifs aux applications publiées par Microsoft sur les machines virtuelles (VM) et les autres nœuds gérés non EC2, vous devez activer le niveau des instances avancées. L'utilisation du niveau d'instance avancé est facturée. La correction d'applications publiées par Microsoft sur des instances Amazon Elastic Compute Cloud (Amazon EC2) n'induit aucuns frais supplémentaires. Pour plus d'informations, consultez Configuration des niveaux d'instance.

Option de mise à jour Windows pour d'« autres produits Microsoft »

Pour permettre à Patch Manager d'installer des correctifs destinés aux applications publiées par Microsoft sur vos nœuds gérés Windows Server, l'option Windows Update Me communiquer les mises à jour d'autres produits Microsoft lorsque je mets à jour Windows doit être activée sur le nœud géré.

Pour savoir comment autoriser cette option sur un nœud géré individuel, consultez Mise à jour d'Office avec Microsoft Update sur le site web du support technique Microsoft.

Pour une flotte de nœuds gérés exécutant Windows Server 2016 ou version ultérieure, vous pouvez activer le paramètre à l'aide d'un objet de politique de groupe (GPO). Dans l'éditeur de gestion des politiques de groupe, accédez à Configuration de l'ordinateur, Modèles administratifs, Composants

Windows, Mises à jour Windows, puis sélectionnez Installer des mises à jour pour d'autres produits Microsoft. Nous recommandons également de configurer l'objet de politique de groupe avec des paramètres supplémentaires qui empêchent les mises à jour automatiques imprévues et les redémarrages en dehors de Patch Manager. Pour de plus amples informations, veuillez vous reporter à Configuration des mises à jour automatiques dans un environnement non Active Directory sur le site web de documentation technique de Microsoft.

Pour une flotte de nœuds gérés exécutant Windows Server 2012 ou 2012 R2, vous pouvez activer l'option à l'aide d'un script, comme décrit dans Enabling and Disabling Microsoft Update in Windows 7 via Script (Activation et désactivation de Microsoft Update dans Windows 7 via un script) sur le site web Microsoft Docs Blog. Par exemple, vous pouvez effectuer les opérations suivantes :

1. Enregistrez le script à partir de l'article de blog dans un fichier.
2. Chargez le fichier dans un compartiment Amazon Simple Storage Service (Amazon S3) ou un autre emplacement accessible.
3. Utilisez la fonctionnalité Run Command d'AWS Systems Manager pour exécuter le script sur vos nœuds gérés en utilisant le document Systems Manager (document SSM) `AWS-RunPowerShellScript` avec une commande semblable à la suivante.

```
Invoke-WebRequest `
    -Uri "https://s3.aws-api-domain/DOC-EXAMPLE-BUCKET/script.vbs" `
    -Outfile "C:\script.vbs" cscript c:\script.vbs
```

Exigences de paramètres minimales

Pour inclure des applications publiées par Microsoft dans votre référentiel de correctifs personnalisé, vous devez, au minimum, spécifier le produit auquel vous souhaitez appliquer un correctif. La commande de l'AWS Command Line Interface (AWS CLI) suivante illustre les exigences minimales pour appliquer un correctif à un produit, comme Microsoft Office 2016.

Linux & macOS

```
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 Server

```
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 & macOS

```
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 Server

```
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}]"
```

> **Note**
>
> Si vous recevez un message d'erreur sur un défaut d'appariement de produits et de familles, veuillez consulter Problème : familles de produits/paires de produits dépareillées pour obtenir de l'aide afin de résoudre le problème.

Utilisation de Kernel Live Patching sur des nœuds gérés 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 profitant d'une infrastructure sécurisée et à jour. Kernel Live Patching est pris en charge sur les instances Amazon EC2, sur les appareils Core AWS IoT Greengrass et sur les machines virtuelles sur site qui exécutent Amazon Linux 2.

Pour obtenir des informations générales Kernel Live Patchingsur Amazon Linux 2 Kernel Live Patching, consultez le guide de l'utilisateur Amazon EC2.

Après avoir Kernel Live Patching activé un nœud géré Amazon Linux 2, vous pouvez utiliser Patch Manager une fonctionnalité de AWS Systems Manager pour appliquer des correctifs dynamiques du noyau au nœud géré. L'utilisation de Patch Manager est une alternative à l'utilisation de flux de travail yum existants sur le nœud pour appliquer les mises à jour.

Avant de commencer

Pour utiliser Patch Manager afin d'appliquer des correctifs à chaud du noyau sur vos nœuds gérés Amazon Linux 2, assurez-vous que vos nœuds sont basés sur la bonne architecture et la bonne version du noyau. Pour plus d'informations, consultez la section Configurations prises en charge et prérequis dans le guide de l'utilisateur Amazon EC2.

Rubriques

À propos de Kernel Live Patching et Patch Manager

Mise à jour de la version du noyau

> Il n'est pas nécessaire de redémarrer un nœud géré après avoir appliqué un correctif à chaud du noyau. Cependant, AWS fournit des correctifs dynamiques du noyau pour une version du noyau Amazon Linux 2 jusqu'à trois mois après sa sortie. 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 nœud au moins une fois tous les trois mois afin de demander la mise à jour de la version du noyau.

Désinstallation des correctifs live du noyau

Les correctifs live du noyau ne peuvent pas être désinstallés à l'aide de Patch Manager. 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 Désactiver Kernel Live Patching en utilisant Run Command.

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. La version la plus récente est alors signalée comme `Installed`, tandis que le nœud géré est signalé comme `Compliant`. Cependant, aucune heure d'installation n'est signalée pour la version plus récente du noyau.

Un correctif live du noyau, plusieurs CVE

Si un patch actif du noyau répond à plusieurs CVE et que ces CVE ont différentes valeurs de classification et de sévérité, seules la classification et la sévérité les plus élevées parmi les CVE sont signalées pour le patch.

Le reste de cette section explique comment utiliser Patch Manager pour appliquer les correctifs à chaud du noyau aux nœuds gérés qui répondent à ces exigences.

Comment ça marche

AWS publie deux types de correctifs dynamiques du noyau pour Amazon Linux 2 : des mises à jour de sécurité et des corrections de bogues. Pour appliquer ces types de correctifs, vous utilisez un document de référentiel de correctifs qui cible uniquement les classifications et les sévérités répertoriées dans le tableau suivant.

Classification	Sévérité
`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 nœuds gérés Amazon Linux 2 sur lesquels Kernel Live Patching est activé. Les mises à jour à chaud du noyau seront alors appliquées.

> **Note**
>
> La `AWS-AmazonLinux2DefaultPatchBaseline` configuration indique une période d'attente de 7 jours après la publication ou la dernière mise à jour d'un correctif avant son installation automatique. Si vous ne voulez pas attendre 7 jours pour que les correctifs en direct du noyau soient automatiquement approuvés, vous pouvez créer et utiliser une base de correctifs personnalisée. Dans votre référentiel de correctifs, vous pouvez spécifier une période d'attente d'approbation automatique nulle ou plus courte ou plus longue. Pour plus d'informations, consultez Utilisation des référentiels de correctifs personnalisés.

Nous vous recommandons la stratégie suivante pour appliquer les mises à jour à chaud du noyau sur vos nœuds gérés :

1. Activez Kernel Live Patching sur vos nœuds gérés Amazon Linux 2.
2. UtilisezRun Command, une fonctionnalité de AWS Systems Manager, pour exécuter une `Scan` opération sur vos nœuds gérés à l'aide de la ligne de base de correctifs prédéfinie `AWS-AmazonLinux2DefaultPatchBaseline` ou personnalisée qui cible également uniquement les `Security` mises à jour dont la gravité est classée comme `Critical` et`Important`, et la `Bugfix` gravité de`All`.
3. Utilisez Conformité, une fonctionnalité de AWS Systems Manager, pour vérifier si une non-conformité en matière de correctifs est signalée pour l'un des nœuds gérés qui ont été scannés. Si tel est le cas, consultez les détails de conformité du nœud pour déterminer s'il manque des correctifs à chaud du noyau dans le nœud géré.
4. Pour installer les correctifs live du noyau manquants, utilisez Run Command 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 le nœud géré si d'autres types de correctifs installés sur celui-ci l'exigent, ou si vous souhaitez procéder à une mise à jour vers un noyau plus récent. Dans ces cas, sélectionnez plutôt l'option de redémarrage `RebootIfNeeded`.

5. Revenez à Conformité pour vérifier que les correctifs live du noyau ont été installés.

Activez Kernel Live Patching en utilisant Run Command

Pour activer Kernel Live Patching, vous pouvez exécuter des commandes `yum` sur vos nœuds gérés, ou utiliser Run Command et un document Systems Manager (document SSM) créé par vos soins.

Pour plus d'informations sur l'activation Kernel Live Patching en exécutant des `yum` commandes directement sur le nœud géré, consultez la section Activer Kernel Live Patching dans le guide de l'utilisateur Amazon EC2.

> **Note**
>
> Lorsque vous activez Kernel Live Patching, si le noyau exécuté sur le nœud géré est antérieur à `kernel-4.14.165-131.185.amzn2.x86_64` (version minimale prise en charge), le processus installe la dernière version disponible du noyau et redémarre le nœud géré. Si le nœud exécute déjà `kernel-4.14.165-131.185.amzn2.x86_64` ou une version ultérieure, le processus n'installe pas de version plus récente et ne redémarre pas le nœud.

Pour activer Kernel Live Patching en utilisant Run Command (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Run Command.
3. Sélectionnez Run Command (Exécuter la commande).
4. Dans la liste Documents de commande, sélectionnez le document SSM personnalisé `AWS-ConfigureKernelLivePatching`.

5. Dans la section Command parameters (Paramètres de commande), indiquez si vous souhaitez redémarrer les nœuds gérés 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 commande à partir de la console.
7. Cliquez sur Exécuter.

Pour activer Kernel Live Patching (AWS CLI)

- Exécutez la commande suivante sur votre machine locale.

  Linux & macOS

  ```
  aws ssm send-command \
      --document-name "AWS-ConfigureKernelLivePatching" \
      --parameters "EnableOrDisable=Enable" \
      --targets "Key=instanceids,Values=instance-id"
  ```

  Windows Server

  ```
  aws ssm send-command ^
      --document-name "AWS-ConfigureKernelLivePatching" ^
      --parameters "EnableOrDisable=Enable" ^
      --targets "Key=instanceids,Values=instance-id"
  ```

  Remplacez *instance-id* par l'ID du nœud géré Amazon Linux 2 sur lequel vous souhaitez activer la fonction, par exemple, i-02573cafcfEXAMPLE. Pour activer la fonction sur plusieurs nœuds gérés, 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 obtenir des informations sur les autres options possibles avec la commande, veuillez consulter send-command dans la Référence des commandes AWS CLI .

Application des correctifs à chaud du noyau à l'aide de Run Command

Pour appliquer des correctifs à chaud du noyau, vous pouvez exécuter des commandes `yum` sur vos nœuds gérés, ou utiliser Run Command et le document SSM `AWS-RunPatchBaseline`.

Pour plus d'informations sur l'application de correctifs dynamiques du noyau en exécutant des `yum` commandes directement sur le nœud géré, consultez la section Appliquer des correctifs dynamiques du noyau dans le guide de l'utilisateur Amazon EC2.

Pour appliquer des correctifs live du noyau à l'aide de Run Command (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Run Command.
3. Sélectionnez Run Command (Exécuter la commande).
4. Dans la liste Command document (Document Command), sélectionnez un document `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 (Opération), sélectionnez `Scan`. Dans Reboot Option (Option de redémarrage), sélectionnez `NoReboot` si vous ne souhaitez pas redémarrer vos nœuds gérés à l'issue de cette opération. Une fois l'opération terminée, vous pouvez vérifier les nouveaux correctifs et l'état de conformité dans Compliance.
   - Si vous avez déjà vérifié la conformité des correctifs et que vous êtes prêt à appliquer les correctifs live du noyau disponibles, pour Opération, sélectionnez `Install`. Dans Reboot Option (Option de redémarrage), sélectionnez `NoReboot` si vous ne souhaitez pas redémarrer vos nœuds gérés à l'issue de cette opération.
6. Pour de plus amples informations sur l'utilisation des contrôles restants de cette page, veuillez consulter Exécution des commande à partir de la console.
7. Cliquez sur Exécuter.

Pour appliquer des correctifs live du noyau à l'aide de Run Command (AWS CLI)

1. Pour effectuer une opération `Scan` avant de vérifier vos résultats dans Compliance, exécutez la commande suivante à partir de votre machine locale.

Linux & macOS

```
aws ssm send-command \
    --document-name "AWS-RunPatchBaseline" \
    --targets "Key=InstanceIds,Values=instance-id" \
    --parameters '{"Operation":["Scan"],"RebootOption":["RebootIfNeeded"]}'
```

Windows Server

```
aws ssm send-command ^
    --document-name "AWS-RunPatchBaseline" ^
    --targets "Key=InstanceIds,Values=instance-id" ^
    --parameters {\"Operation\":[\"Scan\"],\"RebootOption\":[\"RebootIfNeeded
\"]}
```

Pour obtenir des informations sur les autres options possibles avec la commande, veuillez consulter send-command dans la Référence des commandes AWS CLI .

2. Pour effectuer une opération `Install` après avoir vérifié vos résultats dans Compliance, exécutez la commande suivante à partir de votre machine locale.

Linux & macOS

```
aws ssm send-command \
    --document-name "AWS-RunPatchBaseline" \
    --targets "Key=InstanceIds,Values=instance-id" \
    --parameters '{"Operation":["Install"],"RebootOption":["NoReboot"]}'
```

Windows Server

```
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* par l'ID du nœud géré Amazon Linux 2 sur lequel vous souhaitez appliquer des correctifs à chaud du noyau, par exemple

i-02573cafcfEXAMPLE. Pour activer la fonction sur plusieurs nœuds gérés, 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 obtenir des informations sur les autres options possibles avec ces commandes, veuillez consulter send-command dans la Référence des commandes AWS CLI .

Désactiver Kernel Live Patching en utilisant Run Command

Pour désactiver Kernel Live Patching, vous pouvez exécuter des commandes `yum` sur vos nœuds gérés, ou utiliser Run Command et un document SSM `AWS-ConfigureKernelLivePatching` personnalisé.

> **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 fonction n'est pas nécessaire.

Pour plus d'informations sur la désactivation Kernel Live Patching en exécutant `yum` des commandes directement sur le nœud géré, consultez la section Activer Kernel Live Patching dans le guide de l'utilisateur Amazon EC2.

> **Note**
>
> Lorsque vous désactivez Kernel Live Patching, le processus désinstalle le plugin Kernel Live Patching, puis redémarre le nœud géré.

Pour désactiver Kernel Live Patching en utilisant Run Command (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Run Command.
3. Sélectionnez Run Command (Exécuter la commande).

4. Dans la liste Command document (Document Command), sélectionnez un document `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 commande à partir de la console.
7. Cliquez sur Exécuter.

Pour désactiver Kernel Live Patching (AWS CLI)

- Exécutez une commande similaire à la suivante.

  Linux & macOS

  ```
  aws ssm send-command \
      --document-name "AWS-ConfigureKernelLivePatching" \
      --targets "Key=instanceIds,Values=instance-id" \
      --parameters "EnableOrDisable=Disable"
  ```

  Windows Server

  ```
  aws ssm send-command ^
      --document-name "AWS-ConfigureKernelLivePatching" ^
      --targets "Key=instanceIds,Values=instance-id" ^
      --parameters "EnableOrDisable=Disable"
  ```

  Remplacez *instance-id* par l'ID du nœud géré Amazon Linux 2 sur lequel vous souhaitez désactiver la fonction, par exemple i-02573cafcfEXAMPLE. Pour désactiver la fonction sur plusieurs nœuds gérés, 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 obtenir des informations sur les autres options possibles avec la commande, veuillez consulter send-command dans la Référence des commandes AWS CLI .

Utilisation de Patch Manager (console)

Pour utiliser Patch Manager, une fonctionnalité de AWS Systems Manager, procédez comme suit. Ces tâches sont décrites plus en détail dans cette section.

1. Vérifiez que le référentiel 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 un référentiel de correctifs qui définit un ensemble de correctifs standard pour ce type de nœud géré, et définissez-le comme référentiel par défaut.
2. Organisez les nœuds gérés en groupes de correctifs à l'aide de balises Amazon Elastic Compute Cloud (Amazon EC2) (facultatif, mais recommandé).
3. Effectuez l'une des actions suivantes :
   - (Recommandé) Configurez une politique de correctifs dans Quick Setup, une fonctionnalité de Systems Manager qui vous permet d'installer les correctifs manquants selon une planification pour l'ensemble d'une organisation, un sous-ensemble d'unités organisationnelles ou un seul Compte AWS. Pour de plus amples informations, veuillez consulter Configuration des correctifs de l'organisation Patch Manager.
   - Créez une fenêtre de maintenance qui utilise le document Systems Manager (document SSM) `AWS-RunPatchBaseline` dans un type de tâche Run Command. Pour de plus amples informations, veuillez consulter Démonstration : création d'une fenêtre de maintenance pour l'application des correctifs (console).
   - Exécutez `AWS-RunPatchBaseline` manuellement dans une opération Run Command. Pour de plus amples informations, veuillez consulter Exécution des commande à partir de la console.
   - Appliquez manuellement des correctifs aux nœuds à la demande à l'aide de la fonctionnalité Patch now (Appliquer les correctifs maintenant). Pour de plus amples informations, veuillez consulter Application de correctifs sur les nœuds gérés à la demande.
4. Surveillez l'application des correctifs pour vérifier la conformité et enquêter sur les défaillances.

Rubriques

- Affichage des correctifs disponibles
- Utilisation des groupes de correctifs
- Utilisation des paramètres Patch Manager

Création d'une politique de correctif

Une politique de correctifs est une configuration que vous définissez à l'aide de Quick Setup, une fonctionnalité d'AWS Systems Manager. Les politiques de correctif fournissent un contrôle plus étendu et plus centralisé de vos opérations d'application de correctifs qu'avec les autres méthodes. Une politique de correctifs définit la planification et le référentiel à utiliser lors de l'application automatique de correctifs à vos nœuds et applications.

Pour plus d'informations, consultez les rubriques suivantes :

- Utilisation des stratégies de correctifs Quick Setup
- Configuration des correctifs de l'organisation Patch Manager

Affichage des résumés du tableau de bord des correctifs

L'onglet Tableau de bord vous Patch Manager fournit une vue récapitulative dans la console que vous pouvez utiliser pour surveiller vos opérations d'application de correctifs dans une vue consolidée. Patch Managerest une capacité de AWS Systems Manager. L'onglet Dashboard (Tableau de bord) vous permet de visualiser les éléments suivants :

- Un instantané du nombre de nœuds gérés qui sont conformes et non conformes aux règles d'application de correctifs.
- Un instantané de l'ancienneté des résultats de conformité de vos nœuds gérés en matière de correctifs.
- Un décompte lié du nombre de nœuds gérés non conformes pour chacun des motifs courants de non-conformité.
- Une liste liée des opérations d'application de correctifs les plus récentes.
- Une liste liée des tâches récurrentes d'application de correctifs qui ont été configurées.

Pour afficher les résumés du tableau de bord des correctifs

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Patch Manager.
3. Sélectionnez l'onglet Dashboard (Tableau de bord).
4. Faites défiler jusqu'à la section contenant les données récapitulatives à afficher :
   - Gestion des instances Amazon EC2
   - Résumé de conformité
   - Nombre de cas de non-conformité
   - Rapports de conformité
   - Opérations non basées sur des politiques de correctif
   - Tâches récurrentes non basées sur des politiques de correctif

Utilisation des rapports de conformité des correctifs

Les informations contenues dans les rubriques suivantes vous aideront à générer et à utiliser les rapports de conformité des correctifs dans la fonctionnalité Patch Manager d'AWS Systems Manager.

Les informations des rubriques suivantes s'appliquent, quels que soient la méthode ou le type de configuration que vous utilisez pour vos opérations d'application de correctifs :

- Une politique de correctifs configurée dans Quick Setup
- Une option de gestion des hôtes configurée dans Quick Setup
- Une fenêtre de maintenance pour exécuter un correctif `Scan` ou une tâche `Install`
- Une opération Patch now (Appliquer les correctifs maintenant) à la demande

> **Important**
>
> Si vous avez mis en place plusieurs types d'opérations pour analyser la conformité de vos instances aux correctifs, veuillez noter que chaque analyse remplace les données de conformité aux correctifs des analyses précédentes. Par conséquent, vous pourriez obtenir des résultats inattendus en ce qui concerne vos données de conformité aux correctifs. Pour

de plus amples informations, veuillez consulter Éviter les remplacements involontaires des données de conformité aux correctifs.
Pour vérifier quel référentiel de correctifs a été utilisé pour générer les dernières informations de conformité, accédez à l'onglet Rapports de conformité dans Patch Manager, localisez la ligne correspondant au nœud géré qui vous intéresse, puis choisissez l'ID de référentiel dans la colonne ID de référentiel utilisé.

Rubriques

Affichage des résultats de la conformité des correctifs

Utilisez ces procédures pour afficher les informations de conformité des correctifs sur vos nœuds gérés.

Cette procédure s'applique aux opérations d'application de correctifs qui utilisent le document `AWS-RunPatchBaseline`. Pour obtenir des informations sur l'affichage des informations de conformité des correctifs pour les opérations d'application de correctifs qui utilisent le document `AWS-RunPatchBaselineAssociation`, veuillez consulter Identification des nœuds gérés non conformes.

> **Note**
>
> Les opérations de numérisation des correctifs pour le `AWS-RunPatchBaselineAssociation` document Quick Setup et son Explorer utilisation. Quick Setupet Explorer sont toutes deux des fonctionnalités de AWS Systems Manager.

Identification de la solution de correctif pour un problème CVE spécifique (Linux)

Pour de nombreux systèmes d'exploitation Linux, les résultats de conformité des correctifs indiquent quels problèmes signalés sur un bulletin Common Vulnerabilities and Exposure (CVE) sont résolus

par quels correctifs. Ces informations peuvent vous aider à déterminer à quel point il est urgent d'installer un correctif manquant ou défaillant.

Les détails CVE sont inclus pour les versions prises en charge des types de système d'exploitation suivants :

- AlmaLinux
- Amazon Linux 1
- Amazon Linux 2
- Amazon Linux 2022
- Amazon Linux 2023
- Oracle Linux
- Red Hat Enterprise Linux (RHEL)
- Rocky Linux
- SUSE Linux Enterprise Server (SLES)

> **Note**
>
> Par défaut, CentOS et CentOS Stream ne fournissent pas d'informations CVE relatives aux mises à jour. Vous pouvez toutefois autoriser cette prise en charge en utilisant des référentiels tiers tels que le référentiel EPEL (Extra Packages for Enterprise Linux) publié par Fedora. Pour obtenir des informations, veuillez consulter EPEL sur le Wiki Fedora. Actuellement, les valeurs d'identifiant CVE ne sont signalées que pour les correctifs dont le statut est `Missing` ou`Failed`.

Vous pouvez aussi ajouter des ID de CVE à vos listes de correctifs approuvés ou rejetés dans vos référentiels de correctifs, si la situation et vos objectifs d'application de correctifs le justifient.

Pour obtenir des informations sur l'utilisation des listes de correctifs approuvés et de correctifs rejetés, veuillez consulter les rubriques suivantes :

- Utilisation des référentiels de correctifs personnalisés
- À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés
- Fonctionnement des règles de référence de correctif sur les systèmes basés sur Linux
- Installation des correctifs

> **Note**
>
> Dans certains cas, Microsoft publie des correctifs pour les applications qui ne spécifient pas de date et d'heure de mise à jour. La date et l'heure `01/01/1970` sont alors fournies par défaut.

Affichage des résultats de conformité de l'application de correctifs

Utilisez les procédures suivantes pour afficher les données de conformité dans la console AWS Systems Manager .

> **Note**
>
> Pour obtenir des informations sur la génération de rapports de conformité des correctifs qui sont téléchargés dans un compartiment Amazon Simple Storage Service (Amazon S3), veuillez consulter Génération de rapports de conformité des correctifs .csv.

Pour afficher les résultats de conformité des correctifs

1. Effectuez l'une des actions suivantes :

   Option 1 (recommandée) — Naviguez à partir Patch Manager d'une fonctionnalité de AWS Systems Manager :

   - Dans le panneau de navigation, sélectionnez Patch Manager.
   - Sélectionnez l'onglet Compliance reporting (Rapports de conformité).
   - Dans la zone Détails des correctifs du nœud, choisissez l'ID du nœud géré pour lequel vous souhaitez consulter les résultats de conformité des correctifs.
   - Dans la zone Détails, dans la liste des propriétés, sélectionnez Patches.

   Option 2 — Naviguez depuis Conformité, une fonctionnalité permettant de AWS Systems Manager :

   - Dans le panneau de navigation, sélectionnez Compliance (Conformité).

- Pour Récapitulatif des ressources de conformité, sélectionnez un nombre dans la colonne réservée aux types de ressources d'application de correctifs à consulter, tels que Ressources non conformes.
- Ci-dessous, dans la liste des ressources, choisissez l'ID du nœud géré pour lequel vous souhaitez consulter les résultats de conformité des correctifs.
- Dans la zone Détails, dans la liste des propriétés, sélectionnez Patches.

Option 3 — Naviguer à partir Fleet Manager d'une fonctionnalité de AWS Systems Manager.

- Dans le panneau de navigation, sélectionnez Fleet Manager.
- Dans la zone Instances gérées, choisissez l'ID du nœud géré pour lequel vous souhaitez consulter les résultats de conformité des correctifs.
- Dans la zone Détails, dans la liste des propriétés, sélectionnez Patches.

2. (Facultatif) Dans la zone Rechercher
(), sélectionnez parmi les filtres disponibles.

   Par exemple, pour Red Hat Enterprise Linux (RHEL), sélectionnez l'un des éléments suivants :

   - Nom
   - Classification
   - État
   - Sévérité

   Pour Windows Server, sélectionnez parmi les options suivantes :

   - Ko
   - Classification
   - État
   - Sévérité

3. Sélectionnez l'une des valeurs disponibles pour le type de filtre choisi. Par exemple, si vous avez choisi État, choisissez désormais un état de conformité tel que InstalledPendingRebootÉchec ou Manquant.

> **Note**
>
> Actuellement, les valeurs d'identifiant CVE ne sont signalées que pour les correctifs dont le statut est `Missing` ou`Failed`.

4. En fonction de l'état de conformité du nœud géré, vous pouvez choisir l'action à entreprendre pour remédier aux nœuds non conformes.

   Par exemple, vous pouvez choisir d'appliquer immédiatement des correctifs à vos nœuds gérés non conformes. Pour obtenir des informations sur l'application de correctifs sur les nœuds gérés à la demande, consultez Application de correctifs sur les nœuds gérés à la demande.

   Pour plus d'informations sur les états de conformité des correctifs, consultez Comprendre les valeurs d'état de conformité des correctifs.

Génération de rapports de conformité des correctifs .csv

Vous pouvez utiliser la AWS Systems Manager console pour générer des rapports de conformité des correctifs qui sont enregistrés sous forme de fichier .csv dans un bucket Amazon Simple Storage Service (Amazon S3) de votre choix. Vous pouvez générer un rapport à la demande unique ou planifier la génération automatique des rapports.

Les rapports peuvent être générés pour un seul nœud géré ou pour tous les nœuds gérés de votre choix Compte AWS et Région AWS. Un rapport portant sur un nœud individuel contient des détails complets, notamment les ID des correctifs liés à un nœud non conforme. Un rapport portant sur tous les nœuds gérés, quant à lui, ne contient que des informations sommaires ainsi que le nombre de correctifs liés aux nœuds non conformes.

Une fois le rapport généré, vous pouvez utiliser un outil tel QuickSight qu'Amazon pour importer et analyser les données. Amazon QuickSight est un service de business intelligence (BI) que vous pouvez utiliser pour explorer et interpréter des informations dans un environnement visuel interactif. Pour plus d'informations, consultez le guide de QuickSight l'utilisateur Amazon.

> **Note**
>
> Lorsque vous créez un référentiel de correctifs personnalisé, vous pouvez spécifier un niveau de sévérité de conformité pour les correctifs approuvés par ce référentiel de correctifs, tel que `Critical` ou `High`. Si l'état des correctifs d'un correctif approuvé est indiqué `Missing`, le

niveau de sévérité de conformité global indiqué pour le référentiel de correctifs est le niveau de sévérité que vous avez spécifié.

Vous pouvez aussi spécifier une rubrique Amazon Simple Notification Service (Amazon SNS) à utiliser pour envoyer des notifications lorsqu'un rapport est généré.

Rôles de service pour la génération de rapports de conformité des correctifs

La première fois que vous générez un rapport, Systems Manager crée un rôle responsable Automation nommé `AWS-SystemsManager-PatchSummaryExportRole` à utiliser pour le processus d'exportation vers S3.

Note

Si vous exportez des données de conformité vers un compartiment S3 chiffré, vous devez mettre à jour la politique de AWS KMS clé associée afin de fournir les autorisations nécessaires pour`AWS-SystemsManager-PatchSummaryExportRole`. Par exemple, ajoutez une autorisation similaire à celle-ci à la AWS KMS politique de votre compartiment S3 :

```
{
    "Effect": "Allow",
    "Action": [
        "kms:GenerateDataKey"
    ],
    "Resource": "role-arn"
}
```

Remplacez *`role-arn`* par l'Amazon Resource Name (ARN) créé dans votre compte, au format `arn:aws:iam::111222333444:role/service-role/AWS-SystemsManager-PatchSummaryExportRole`.
Pour plus d'informations, consultez Politiques de clé dans AWS KMS dans le Guide du développeur AWS Key Management Service .

La première fois que vous générez un rapport selon un calendrier, Systems Manager crée un autre rôle de service nommé`AWS-EventBridge-Start-SSMAutomationRole`, ainsi que le rôle de service `AWS-SystemsManager-PatchSummaryExportRole` (s'il n'est

pas déjà créé) à utiliser pour le processus d'exportation. `AWS-EventBridge-Start-SSMAutomationRole`permet EventBridge à Amazon de démarrer une automatisation à l'aide du runbook AWS- ExportPatchReportTo S3.

Nous vous déconseillons de tenter de modifier ces politiques et ces rôles. Cela pourrait entraîner l'échec de la génération de rapports de conformité des correctifs. Pour plus d'informations, consultez Résolution des problèmes liés à la génération de rapports de conformité des correctifs.

Rubriques

Qu'est-ce qu'un rapport de conformité des correctifs généré ?

Cette rubrique fournit des informations sur les types de contenu inclus dans les rapports de conformité des correctifs qui sont générés et téléchargés dans un compartiment S3 spécifié.

Format de rapport pour un nœud géré individuel

Un rapport généré pour un nœud géré individuel fournit à la fois des informations sommaires et détaillées.

Télécharger un exemple de rapport (pour un nœud individuel)

Les informations sommaires fournies pour un nœud géré individuel sont les suivantes :

- Index
- ID d'instance
- Instance name
- Adresse IP d'instance
- Nom de la plateforme
- Version de plateforme

- Version de SSM Agent
- Référentiel de correctifs
- Groupe de correctifs
- Statut de conformité
- Sévérité de conformité
- Nombre de correctifs de sévérité critique non conformes
- Nombre de correctifs de sévérité élevée non conformes
- Nombre de correctifs de sévérité moyenne non conformes
- Nombre de correctifs de sévérité faible non conformes
- Nombre de correctifs de sévérité informationnelle non conformes
- Nombre de correctifs de sévérité non spécifiée non conformes

Les informations détaillées fournies pour un nœud géré individuel sont les suivantes :

- Index
- ID d'instance
- Instance name
- Nom du correctif
- ID de la KB/ID du patch
- État du correctif
- Heure du dernier rapport
- Niveau de conformité
- Sévérité du correctif
- Classification des correctifs
- ID CVE
- Référentiel de correctifs
- URL des journaux
- Adresse IP d'instance
- Nom de la plateforme
- Version de plateforme

- Version de SSM Agent

> **Note**
>
> Lorsque vous créez un référentiel de correctifs personnalisé, vous pouvez spécifier un niveau de sévérité de conformité pour les correctifs approuvés par ce référentiel de correctifs, tel que `Critical` ou `High`. Si l'état des correctifs d'un correctif approuvé est indiqué `Missing`, le niveau de sévérité de conformité global indiqué pour le référentiel de correctifs est le niveau de sévérité que vous avez spécifié.

Format de rapport pour tous les nœuds gérés

Un rapport généré pour tous les nœuds gérés ne fournit que des informations sommaires.

Télécharger un exemple de rapport (pour tous les nœuds gérés)

Les informations sommaires fournies pour tous les nœuds gérés sont les suivantes :

- Index
- ID d'instance
- Instance name
- Adresse IP d'instance
- Nom de la plateforme
- Version de plateforme
- Version de SSM Agent
- Référentiel de correctifs
- Groupe de correctifs
- Statut de conformité
- Sévérité de conformité
- Nombre de correctifs de sévérité critique non conformes
- Nombre de correctifs de sévérité élevée non conformes
- Nombre de correctifs de sévérité moyenne non conformes
- Nombre de correctifs de sévérité faible non conformes
- Nombre de correctifs de sévérité informationnelle non conformes

- Nombre de correctifs de sévérité non spécifiée non conformes

Génération de rapports de conformité des correctifs pour un nœud géré individuel

Procédez comme suit pour générer un rapport sommaire sur les correctifs relatifs à un nœud géré individuel dans votre Compte AWS. Le rapport généré pour un nœud géré individuel fournit des détails sur chaque correctif non conforme, notamment son nom et son ID.

Pour générer des rapports de conformité des correctifs pour un nœud géré individuel

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Patch Manager.
3. Sélectionnez l'onglet Compliance reporting (Rapports de conformité).
4. Cliquez sur le bouton correspondant à la ligne du nœud géré pour lequel vous souhaitez générer un rapport, puis sélectionnez View detail (Afficher les détails).
5. Dans la section Patch summary (Récapitulatif des correctifs), sélectionnez Export to S3 (Exporter vers S3).
6. Pour Nom du rapport, saisissez un nom qui vous aidera à identifier le rapport ultérieurement.
7. Pour Fréquence de génération de rapports, sélectionnez l'une des options suivantes :
   - À la demande : pour créer un rapport unique. Passez à l'étape 9.
   - Planifiée : spécifie une planification récurrente pour la génération automatique de rapports. Passez à l'étape 8.
8. Pour Type de programme, spécifiez une expression rate, par exemple tous les 3 jours, ou fournissez une expression cron pour définir la fréquence du rapport.

   Pour plus d'informations sur les expressions cron, consultez Référence : Expressions Cron et Rate pour Systems Manager.
9. Pour Nom du compartiment, sélectionnez le nom du compartiment S3 dans lequel vous voulez stocker les fichiers de rapport .csv.

   > **Important**
   >
   > Si vous travaillez dans un Région AWS compartiment lancé après le 20 mars 2019, vous devez sélectionner un compartiment S3 dans cette même région. Les régions lancées après cette date ont été désactivées par défaut. Pour plus d'informations et une liste

de ces régions, veuillez consulter la rubrique Activation d'une région dans la Référence générale d'Amazon Web Services.

10. (Facultatif) Pour envoyer des notifications lorsque le rapport est généré, développez la section SNS topic (Rubrique SNS), puis sélectionnez une rubrique Amazon SNS existante dans SNS topic Amazon Resource Name (ARN) (Amazon Resource Name (ARN) de rubrique SNS).
11. Sélectionnez Submit (Envoyer).

Pour obtenir des informations sur l'affichage d'un historique des rapports générés, veuillez consulter Affichage de l'historique de génération de rapports de conformité des correctifs.

Pour obtenir des informations sur l'affichage des détails relatifs aux calendriers de génération de rapports que vous avez créés, veuillez consulter Affichage des calendriers de rapports de conformité des correctifs.

Génération de rapports de conformité des correctifs pour tous les nœuds gérés

Procédez comme suit pour générer un rapport sommaire sur les correctifs relatifs à tous les nœuds gérés de votre Compte AWS. Le rapport portant sur tous les nœuds gérés désigne les nœuds qui ne sont pas conformes et le nombre de correctifs non conformes. Il ne fournit pas les noms ou autres identifiants des correctifs. Pour plus de détails, vous pouvez générer un rapport de conformité des correctifs portant sur un nœud géré individuel. Pour plus d'informations, consultez Génération de rapports de conformité des correctifs pour un nœud géré individuel plus haut dans cette rubrique.

Pour générer des rapports de conformité des correctifs pour tous les nœuds gérés

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Patch Manager.
3. Sélectionnez l'onglet Compliance reporting (Rapports de conformité).
4. Sélectionnez Exporter vers S3. (Évitez de sélectionner un ID de nœud en premier).
5. Pour Nom du rapport, saisissez un nom qui vous aidera à identifier le rapport ultérieurement.
6. Pour Fréquence de génération de rapports, sélectionnez l'une des options suivantes :

   - À la demande : pour créer un rapport unique. Passez à l'étape 8.
   - Planifiée : spécifie une planification récurrente pour la génération automatique de rapports. Passez à l'étape 7.

7. Pour Type de programme, spécifiez une expression rate, par exemple tous les 3 jours, ou fournissez une expression cron pour définir la fréquence du rapport.

   Pour plus d'informations sur les expressions cron, consultez Référence : Expressions Cron et Rate pour Systems Manager.

8. Pour Nom du compartiment, sélectionnez le nom du compartiment S3 dans lequel vous voulez stocker les fichiers de rapport .csv.

   > **Important**
   >
   > Si vous travaillez dans un Région AWS compartiment lancé après le 20 mars 2019, vous devez sélectionner un compartiment S3 dans cette même région. Les régions lancées après cette date ont été désactivées par défaut. Pour plus d'informations et une liste de ces régions, veuillez consulter la rubrique Activation d'une région dans la Référence générale d'Amazon Web Services.

9. (Facultatif) Pour envoyer des notifications lorsque le rapport est généré, développez la section SNS topic (Rubrique SNS), puis sélectionnez une rubrique Amazon SNS existante dans SNS topic Amazon Resource Name (ARN) (Amazon Resource Name (ARN) de rubrique SNS).
10. Sélectionnez Submit (Envoyer).

Pour obtenir des informations sur l'affichage d'un historique des rapports générés, veuillez consulter Affichage de l'historique de génération de rapports de conformité des correctifs.

Pour obtenir des informations sur l'affichage des détails relatifs aux calendriers de génération de rapports que vous avez créés, veuillez consulter Affichage des calendriers de rapports de conformité des correctifs.

Affichage de l'historique de génération de rapports de conformité des correctifs

Utilisez les informations de cette rubrique pour vous aider à consulter les détails des rapports de conformité des correctifs générés dans votre Compte AWS.

Pour afficher l'historique de génération de rapports de conformité des correctifs

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Patch Manager.

3. Sélectionnez l'onglet Compliance reporting (Rapports de conformité).
4. Sélectionnez Afficher toutes les exportations vers S3, puis sélectionnez l'onglet Historique des exportations.

Affichage des calendriers de rapports de conformité des correctifs

Utilisez les informations de cette rubrique pour vous aider à consulter les détails des calendriers de rapports de conformité des correctifs créés dans votre Compte AWS.

Pour afficher l'historique de génération de rapports de conformité des correctifs

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Patch Manager.
3. Sélectionnez l'onglet Compliance reporting (Rapports de conformité).
4. Sélectionnez View all S3 exports (Afficher toutes les exportations S3), puis l'onglet Report schedule rules (Règles de planification de rapport).

Résolution des problèmes liés à la génération de rapports de conformité des correctifs

Utilisez les informations suivantes pour résoudre les problèmes liés à la génération de rapports de conformité des correctifs dans Patch Manager, une fonctionnalité de AWS Systems Manager.

Rubriques

- Un message signale que la politique AWS-SystemsManager-PatchManagerExportRolePolicy est corrompue
- La suppression des politiques ou des rôles de conformité des correctifs empêche la génération correcte des rapports planifiés.

Un message signale que la politique **`AWS-SystemsManager-PatchManagerExportRolePolicy`** est corrompue

Problème : vous recevez un message d'erreur semblable au suivant, indiquant que la valeur `AWS-SystemsManager-PatchManagerExportRolePolicy` est corrompue :

```
An error occurred while updating the AWS-SystemsManager-PatchManagerExportRolePolicy
policy. If you have edited the policy, you might need to delete the policy, and any
```

```
role that uses it, then try again. Systems Manager recreates the roles and policies
you have deleted.
```

- Solution : utilisez la Patch Manager console ou supprimez AWS CLI les rôles et politiques concernés avant de générer un nouveau rapport de conformité des correctifs.

  Pour supprimer la politique corrompue à l'aide de la console

  1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
  2. Effectuez l'une des actions suivantes :

     Rapports à la demande : si le problème s'est produit lors de la génération d'un rapport à la demande ponctuel, dans le panneau de navigation de gauche, sélectionnez Politiques, recherchez `AWS-SystemsManager-PatchManagerExportRolePolicy`, puis supprimez la politique. Ensuite, sélectionnez Rôles, recherchez `AWS-SystemsManager-PatchSummaryExportRole`, puis supprimez le rôle.

     Rapports planifiés : si le problème s'est produit lors de la génération d'un rapport planifié, dans le panneau de navigation de gauche, sélectionnez Politiques, recherchez `AWS-EventBridge-Start-SSMAutomationRolePolicy` et `AWS-SystemsManager-PatchManagerExportRolePolicy` une par une, et supprimez chaque politique. Ensuite, sélectionnez Rôles, recherchez `AWS-EventBridge-Start-SSMAutomationRole` et `AWS-SystemsManager-PatchSummaryExportRole` un par un, et supprimez chaque rôle.

  Pour supprimer la politique corrompue à l'aide du AWS CLI

  Remplacez les *`valeurs de remplacements`* par l'identifiant de votre compte.

  - Si le problème s'est produit lors de la génération d'un rapport unique à la demande, exécutez les commandes suivantes :

    ```
    aws iam delete-policy --policy-arn arn:aws:iam::account-id:policy/AWS-
    SystemsManager-PatchManagerExportRolePolicy
    ```

    ```
    aws iam delete-role --role-name AWS-SystemsManager-PatchSummaryExportRole
    ```

    Si le problème s'est produit lors de la génération d'un rapport selon une planification, exécutez les commandes suivantes :

```
aws iam delete-policy --policy-arn arn:aws:iam::account-id:policy/AWS-
EventBridge-Start-SSMAutomationRolePolicy
```

```
aws iam delete-policy --policy-arn arn:aws:iam::account-id:policy/AWS-
SystemsManager-PatchManagerExportRolePolicy
```

```
aws iam delete-role --role-name AWS-EventBridge-Start-SSMAutomationRole
```

```
aws iam delete-role --role-name AWS-SystemsManager-PatchSummaryExportRole
```

Après avoir effectué l'une des deux procédures, suivez les étapes pour générer ou planifier un nouveau rapport de conformité des correctifs.

La suppression des politiques ou des rôles de conformité des correctifs empêche la génération correcte des rapports planifiés.

Problème : la première fois que vous générez un rapport, Systems Manager crée un rôle de service et une politique à utiliser pour le processus d'exportation (`AWS-SystemsManager-PatchSummaryExportRole` et `AWS-SystemsManager-PatchManagerExportRolePolicy`). La première fois que vous générez un rapport planifié, Systems Manager crée un autre rôle de service et une autre politique (`AWS-EventBridge-Start-SSMAutomationRole` et `AWS-EventBridge-Start-SSMAutomationRolePolicy`). Ils permettent EventBridge à Amazon de démarrer une automatisation à l'aide du runbook AWS- ExportPatchReportTo S3.

Si vous supprimez l'une de ces politiques ou l'un de ces rôles, les connexions entre votre calendrier et votre compartiment S3 spécifié et la rubrique Amazon SNS peuvent être perdues.

- Solution : pour contourner ce problème, nous vous recommandons de supprimer le calendrier précédent et de créer un calendrier pour remplacer celui qui présentait des problèmes.

Correction des nœuds gérés non conformes avec Patch Manager

Les rubriques de cette section expliquent comment identifier les nœuds gérés qui ne sont pas conformes en matière de correctifs, et comment les mettre en conformité.

Rubriques

- Identification des nœuds gérés non conformes
- Comprendre les valeurs d'état de conformité des correctifs
- Application de correctifs sur des nœuds gérés non conformes

Identification des nœuds gérés non conformes

Les nœuds ut-of-compliance gérés sont identifiés lorsque l'un des deux AWS Systems Manager documents (documents SSM) est exécuté. Ces documents SSM font référence au référentiel de correctifs correspondant à chaque nœud géré dans la fonctionnalité Patch Manager d'AWS Systems Manager. Ils évaluent ensuite l'état des correctifs du nœud géré, puis mettent les résultats de conformité à votre disposition.

Deux documents SSM sont utilisés pour identifier ou mettre à jour les nœuds gérés non conformes : `AWS-RunPatchBaseline` et `AWS-RunPatchBaselineAssociation`. Chacun d'entre eux 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 relatives à l'application de correctifs depuis Patch Manager peuvent être envoyées à AWS Security Hub. Security Hub vous offre une vue complète sur vos alertes de sécurité haute priorité et votre statut de conformité. Il surveille également le statut d'application des correctifs de votre flotte. Pour plus d'informations, consultez Intégration Patch Manager avec AWS Security Hub.

	AWS-RunPatchBaseline	**AWS-RunPatchBaselineAssociation**
Processus qui utilisent le document	Correctifs à la demande : vous pouvez analyser les nœuds gérés ou y appliquer des correctifs à la demande à l'aide de l'option Patch now (Appliquer les correctifs maintenant). Pour plus d'informations, consultez	Gestion des hôtes Quick Setup de Systems Manager : vous pouvez activer une option de configuration de gestion des hôtes dans Quick Setup de sorte à analyser quotidiennement la conformité aux correctifs de vos instances

	AWS-RunPatchBaseline	AWS-RunPatchBaselineAssociation
	Application de correctifs sur les nœuds gérés à la demande. Politiques de correctif Quick Setup de Systems Manager : vous pouvez créer une configuration d'application de correctifs dans Quick Setup, une fonctionnalité d'AWS Systems Manager, capable de rechercher ou d'installer les correctifs manquants selon des planifications distinctes pour l'ensemble d'une organisation, un sous-ensemble d'unités organisationnelles ou un seul Compte AWS. Pour plus d'informations, consultez Configuration des correctifs de l'organisation Patch Manager. Exécution d'une commande : vous pouvez exécuter `AWS-RunPatchBaseline` manuellement dans une opération dans Run Command, une fonctionnalité de AWS Systems Manager. Pour plus d'informations, consultez Exécution des commande à partir de la console.	gérées. Pour plus d'informations, consultez Gestion des hôtes Amazon EC2. Systems Manager Explorer : lorsque vous autorisez Explorer, une fonctionnalité de AWS Systems Manager, il analyse régulièrement vos instances gérées pour vérifier la conformité des correctifs et générer des rapports sur les résultats dans le tableau de bord Explorer.

	AWS-RunPatchBaseline	AWS-RunPatchBaselineAssociation
	Fenêtre de maintenance : vous pouvez créer une fenêtre de maintenance qui utilise le document SSM `AWS-RunPatchBaseline` dans un type de tâche Run Command. Pour plus d'informations, consultez Démonstration : création d'une fenêtre de maintenance pour l'application des correctifs (console).	
Format des données de résultat de l'analyse des correctifs	Une fois que `AWS-RunPatchBaseline` s'exécute, Patch Manager envoie un objet `AWS:PatchSummary` à Inventory, une fonctionnalité de AWS Systems Manager.	Une fois que `AWS-RunPatchBaselineAssociation` s'exécute, Patch Manager envoie un objet `AWS:ComplianceItem` à Systems Manager Inventory.

	AWS-RunPatchBaseline	AWS-RunPatchBaselineAssociation
Affichage des rapports de conformité actuelle dans la console	Vous pouvez afficher des informations de conformité des correctifs pour les processus qui utilisent `AWS-RunPatchBaseline` dans Conformité de la configuration Systems Manager et Utilisation de nœuds gérés. Pour plus d'informations, consultez Affichage des résultats de la conformité des correctifs.	Si vous utilisez Quick Setup pour analyser vos instances gérées pour vérifier la conformité des correctifs, vous pouvez consulter le rapport de conformité dans Systems Manager State Manager, accessible via un bouton Afficher les résultats dans Quick Setup. Si vous utilisez Explorer pour analyser vos instances gérées pour la conformité des correctifs, vous pouvez voir le rapport de conformité dans Explorer et Systems Manager OpsCenter.
Commandes de la AWS CLI pour afficher les résultats de conformité des correctifs	Pour les processus qui utilisent `AWS-RunPatchBaseline`, vous pouvez aussi utiliser les commandes AWS CLI suivantes pour afficher des informations sommaires sur les correctifs d'un nœud géré. • describe-instance-patch-states • describe-instance-patch-states-for-patch-group • describe-patch-group-state	Pour les processus qui utilisent `AWS-RunPatchBaselineAssociation`, vous pouvez aussi utiliser la commande AWS CLI suivante pour afficher les informations récapitulatives concernant les correctifs d'une instance. • list-compliance-items

	AWS-RunPatchBaseline	AWS-RunPatchBaseli neAssociation
Opérations d'application de correctifs	Pour les processus qui utilisent `AWS-RunPa tchBaseline` , vous spécifiez si l'opération doit exécuter une opération `Scan` uniquement ou une opération `Scan and install`. Si votre objectif est d'identif ier les nœuds gérés non conformes, et non de les corriger, contentez-vous d'exécuter une opération `Scan`.	Les processus Quick Setup et Explorer, qui utilisent `AWS-RunPatchBaseli neAssociation` , exécutent uniquement une opération `Scan`.
Plus d'informations	À propos du document SSM `AWS-RunPatchBaseline`	À propos du document SSM `AWS-RunPatchBaseli neAssociation`

Pour obtenir des informations sur les différents états de conformité des correctifs qui peuvent être signalés, veuillez consulter Comprendre les valeurs d'état de conformité des correctifs

Pour obtenir des informations sur la résolution des problèmes de non-conformité des nœuds gérés, consultez Application de correctifs sur des nœuds gérés non conformes.

Comprendre les valeurs d'état de conformité des correctifs

Les informations relatives aux correctifs d'un nœud géré incluent un rapport sur l'état ou le statut de chaque correctif.

> **Note**
>
> Si vous souhaitez attribuer un état de conformité des correctifs spécifique à un nœud géré, vous pouvez utiliser la commande put-compliance-items AWS Command Line Interface (AWS

CLI) ou l'opération PutComplianceItemsAPI. L'attribution d'un état de conformité n'est pas prise en charge dans la console.

Utilisez les informations figurant dans les tableaux suivants pour identifier les raisons pour lesquelles une nœud géré peut ne pas être conforme en matière de correctifs.

Valeurs de conformité des correctifs pour Debian Server, Raspberry Pi OS et Ubuntu Server

Pour Debian Server, Raspberry Pi OS et Ubuntu Server, les règles de classification des packages dans les différents états de conformité sont décrites dans le tableau suivant.

Note

Lorsque vous évaluez les valeurs de statut Installé, Installé Autre et Manquant, n'oubliez pas que : 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'un référentiel de correctifs, les versions des correctifs candidates se limitent aux correctifs inclus dans `trusty-security` (Ubuntu Server 14.04 LTS), `xenial-security` (Ubuntu Server 16.04 LTS), `bionic-security` (Ubuntu Server 18.04 LTS), `focal-security` (Ubuntu Server 20.04 LTS), `groovy-security` (Ubuntu Server 20.10 STR), `jammy-security` (Ubuntu Server 22.04 LTS) ou `debian-security` (Debian Server et Raspberry Pi OS). Si vous cochez la case Inclure les mises à jour non liées à la sécurité, les correctifs provenant d'autres référentiels sont également pris en compte.

État du correctif	Description	Statut de conformité
`INSTALLED`	Le correctif est répertorié dans le référentiel de correctifs et est installé sur le nœud géré. Il a pu être installé manuellem ent par une personne ou automatiquement par Patch Manager lors de l'exécution du document `AWS-RunPa`	Conforme

État du correctif	Description	Statut de conformité
	`tchBaseline` sur le nœud géré.	
`INSTALLED_OTHER`	Le correctif n'est pas inclus dans le référentiel ou n'est pas approuvé par le référentiel, mais il est installé sur le nœud géré. Le correctif a peut-être 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` comme l'action Correctifs rejetés, `Installed_Other` inclut également les correctifs installés mais rejetés.	Conforme

État du correctif	Description	Statut de conformité
INSTALLED_PENDING_REBOOT	INSTALLED_PENDING_REBOOT peut signifier l'une des deux choses suivantes : • L'opération Install de Patch Manager a appliqué le correctif sur le nœud géré, mais le nœud n'a pas été redémarré depuis l'application du correctif . Cela signifie généralement que le paramètre RebootOption a été défini sur NoReboot lors de la dernière exécution du document AWS-RunPatchBaseline sur le nœud géré. Pour plus d'informations, consultez Nom du paramètre: RebootOption . • Un correctif a été installé en dehors de Patch Manager depuis le dernier redémarrage du nœud géré.	Non-Compliant (Non conforme)
INSTALLED_REJECTED	Le correctif est installé sur le nœud géré, mais il figure dans une liste Rejected patches (Correctifs rejetés). Cela signifie généralement que le correctif a été installé avant d'être ajouté à la liste des correctifs rejetés.	Non-Compliant (Non conforme)

État du correctif	Description	Statut de conformité
MISSING	Packages qui sont filtrés via le référentiel, mais ne sont pas encore installés.	Non-Compliant (Non conforme)
FAILED	Packages dont l'installation a échoué pendant l'opération d'application de correctif.	Non-Compliant (Non conforme)

Valeurs de conformité des correctifs pour les autres systèmes d'exploitation

Pour tous les systèmes d'exploitation autres que Debian Server, Raspberry Pi OS 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	Le correctif est répertorié dans le référentiel de correctifs et est installé sur le nœud géré. Il a pu être installé manuellement par une personne ou automatiquement par Patch Manager lors de l'exécution du document `AWS-RunPatchBaseline` sur le nœud.	Conforme
INSTALLED_OTHER [1]	Le correctif ne figure pas dans le référentiel, mais il est installé sur le nœud géré. Il se peut que le correctif ait été installé manuellement ou que le package soit une dépendance obligatoire d'un autre correctif approuvé. Si vous ne spécifiez pas	Conforme

État du correctif	Description	Valeur de conformité
	`Block` comme l'action Correctifs rejetés, `Installed_Other` inclut également les correctifs installés mais rejetés.	
`INSTALLED_REJECTED`	Le correctif est installé sur le nœud géré, mais il figure dans une liste Rejected patches (Correctifs rejetés). Cela signifie généralement que le correctif a été installé avant d'être ajouté à la liste des correctifs rejetés.	Non-Compliant (Non conforme)

État du correctif	Description	Valeur de conformité
INSTALLED_PENDING_REBOOT	INSTALLED_PENDING_REBOOT peut signifier l'une des deux choses suivantes : • L'opération Install de Patch Manager a appliqué le correctif sur le nœud géré, mais le nœud n'a pas été redémarré depuis l'application du correctif . Cela signifie généralement que le paramètre RebootOption a été défini sur NoReboot lors de la dernière exécution du document AWS-RunPatchBaseline sur le nœud géré. Pour plus d'informations, consultez Nom du paramètre: RebootOption . • Un correctif a été installé en dehors de Patch Manager depuis le dernier redémarrage du nœud géré.	Non-Compliant (Non conforme)

État du correctif	Description	Valeur de conformité
MISSING	Le correctif est approuvé dans le référentiel, mais il n'est pas installé sur le nœud géré. 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-Compliant (Non conforme)

État du correctif	Description	Valeur de conformité
NOT_APPLICABLE [1]	Le correctif est approuvé dans le référentiel, mais le service ou la fonction qui l'utilise n'est pas installé sur le nœud géré. Par exemple, un correctif relatif à un service de serveur web tel qu'Internet Informati on Services (IIS) indique NOT_APPLICABLE s'il a été approuvé dans le référentiel, alors que le service web n'est pas installé sur le nœud géré. 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 conformit é est uniquemen t signalé sur les systèmes d'exploit ation Windows Server.	Ne s'applique pas

État du correctif	Description	Valeur de conformité
FAILED	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-Compliant (Non conforme)

[1] Pour les correctifs avec l'état INSTALLED_OTHER et NOT_APPLICABLE, Patch Manager omet certaines données des résultats de la demande en fonction de la commande describe-instance-patches, comme les valeurs pour `Classification` et `Severity`. Ceci est fait pour éviter de dépasser la limite de données pour les nœuds individuels dans Inventory, une fonctionnalité de AWS Systems Manager. Pour voir tous les détails des correctifs, vous pouvez utiliser la commande describe-available-patches.

Application de correctifs sur des nœuds gérés non conformes

La plupart des outils et processus AWS Systems Manager utilisables pour vérifier la conformité des nœuds gérés en matière de correctifs peuvent également être utilisés pour mettre les nœuds en conformité avec les règles d'application de correctifs qui s'y rapportent. Pour mettre les nœuds gérés en conformité, la fonctionnalité Patch Manager d'AWS Systems Manager doit exécuter une opération `Scan and install`. (Si votre objectif est uniquement d'identifier les nœuds gérés non conformes, et non de les corriger, contentez-vous d'exécuter une opération `Scan`. Pour plus d'informations, veuillez consulter la rubrique Identification des nœuds gérés non conformes.)

Installer des correctifs en utilisant Systems Manager

Vous pouvez choisir parmi plusieurs outils pour exécuter une opération `Scan and install`.

- (Recommandé) Configurez une politique de correctifs dans Quick Setup, une fonctionnalité de Systems Manager qui vous permet d'installer les correctifs manquants selon une planification pour l'ensemble d'une organisation, un sous-ensemble d'unités organisationnelles ou un seul Compte AWS. Pour de plus amples informations, veuillez consulter Configuration des correctifs de l'organisation Patch Manager.

- Créez une fenêtre de maintenance qui utilise le document Systems Manager (document SSM) `AWS-RunPatchBaseline` dans un type de tâche Run Command. Pour plus d'informations, consultez Démonstration : création d'une fenêtre de maintenance pour l'application des correctifs (console).
- Exécutez `AWS-RunPatchBaseline` manuellement dans une opération Run Command. Pour plus d'informations, consultez Exécution des commande à partir de la console.

- Installez des correctifs à la demande en utilisant l'option Corriger maintenant. Pour plus d'informations, consultez Application de correctifs sur les nœuds gérés à la demande.

Éviter les remplacements involontaires des données de conformité aux correctifs

Si vous avez mis en place plusieurs types d'opérations pour analyser la conformité de vos instances aux correctifs, chaque analyse remplace les données de conformité aux correctifs des analyses précédentes. Par conséquent, vous pourriez obtenir des résultats inattendus en ce qui concerne vos données de conformité aux correctifs.

Supposons, par exemple, que vous créiez une politique de correctifs qui analyse la conformité aux correctifs chaque jour à 2 h 00, heure locale. Cette politique de correctifs utilise un référentiel de correctifs qui cible les correctifs dont la sévérité est définie sur `Critical`, `Important` et `Moderate`. Ce référentiel de correctifs indique également quelques correctifs spécifiquement rejetés.

Supposons également qu'une fenêtre de maintenance ait déjà été configurée pour analyser le même ensemble de nœuds gérés chaque jour à 4 h 00, heure locale, que vous ne supprimez ni ne désactivez. La tâche de cette fenêtre de maintenance utilise un référentiel de correctifs différent, qui cible uniquement les correctifs dont la sévérité est `Critical` et n'exclut aucun correctif spécifique.

Lorsque cette seconde analyse est effectuée par la fenêtre de maintenance, les données de conformité aux correctifs de la première analyse sont supprimées et remplacées par les données de conformité aux correctifs issues de la seconde analyse.

Par conséquent, nous vous recommandons vivement de n'utiliser qu'une seule méthode automatisée pour analyser et installer vos opérations d'application de correctifs. Si vous configurez des politiques de correctif, vous devez supprimer ou désactiver les autres méthodes d'analyse de la conformité aux correctifs. Pour plus d'informations, consultez les rubriques suivantes :

- Pour supprimer une tâche d'application de correctifs d'une fenêtre de maintenance : Mise à jour de tâches de fenêtre de maintenance ou annulation de leur enregistrement (console)

- Pour supprimer une association State Manager : Suppression d'associations.

Pour désactiver les analyses quotidiennes de conformité aux correctifs dans une configuration de gestion des hôtes, procédez comme suit dans Quick Setup :

1. Dans le panneau de navigation, sélectionnez Quick Setup.
2. Sélectionnez la configuration de gestion des hôtes à mettre à jour.
3. Choisissez Actions, Edit configuration (Actions, Modifier la configuration).
4. Décochez la case Scan instances for missing patches daily (Analyser quotidiennement les instances pour les correctifs manquants).
5. Choisissez Mettre à jour.

> **Note**
>
> L'utilisation de l'option Patch now (Appliquer les correctifs maintenant) pour analyser la conformité d'un nœud géré entraîne également le remplacement des données de conformité aux correctifs.

Application de correctifs sur les nœuds gérés à la demande

L'utilisation de l'option Corriger maintenant dans Patch Manager, une fonctionnalité de AWS Systems Manager, vous permet d'exécuter des opérations d'application de correctifs à la demande depuis la console Systems Manager. Cela signifie que vous n'avez pas à créer de calendrier pour mettre à jour le statut de conformité de vos nœuds gérés ou pour installer des correctifs sur les nœuds non conformes. Vous n'avez pas non plus besoin de permuter la console Systems Manager entre Patch Manager et Maintenance Windows, une fonctionnalité de AWS Systems Manager, pour configurer ou modifier une fenêtre d'application de correctifs planifiée.

L'option Patch now (Appliquer les correctifs maintenant) est particulièrement utile lorsque vous devez appliquer des mises à jour « zéro jour » ou installer d'autres correctifs critiques sur vos nœuds gérés dans les plus brefs délais.

> **Note**
>
> L'application de correctifs à la demande est prise en charge pour une seule Compte AWSRégion AWS paire à la fois. Elle ne peut pas être utilisée avec des opérations

d'application de correctifs basées sur des politiques de correctif. Nous vous recommandons d'utiliser des politiques de correctif pour garantir la conformité de tous vos nœuds gérés. Pour plus d'informations sur l'utilisation des politiques de correctifs, consultez la rubrique Utilisation des stratégies de correctifs Quick Setup.

Rubriques

- Fonctionnement de l'option « Corriger maintenant »
- Exécution de l'option « Corriger maintenant »

Fonctionnement de l'option « Corriger maintenant »

Pour exécuter l'option Corriger maintenant, vous devez spécifier deux paramètres obligatoires seulement :

- La simple recherche des correctifs manquants, ou l'analyse et l'installation des correctifs sur vos nœuds gérés
- Les nœuds gérés sur lesquels exécuter l'opération

Lorsque l'opération Patch now (Appliquer les correctifs maintenant) s'exécute, elle détermine le référentiel de correctifs à utiliser, comme pour les autres opérations d'application de correctifs. Si un nœud géré est associé à un groupe de correctifs, le référentiel de correctifs spécifié pour ce groupe est utilisé. Si le nœud géré n'est associé à aucun groupe de correctifs, l'opération utilise le référentiel de correctifs défini par défaut pour le type de système d'exploitation du nœud géré. Il peut s'agir d'un référentiel prédéfini ou du référentiel personnalisé que vous avez défini par défaut. Pour plus d'informations sur la sélection du référentiel de correctifs, consultez À propos des groupes de correctifs.

Parmi les options que vous pouvez spécifier pour l'opération Patch now (Appliquer les correctifs maintenant) figurent le choix du moment, ou de l'opportunité, de redémarrer les nœuds gérés après l'application de correctifs, la spécification d'un compartiment Amazon Simple Storage Service (Amazon S3) pour stocker les données de journal de l'opération d'application de correctifs, et l'exécution de documents Systems Manager (documents SSM) en tant que hooks de cycle de vie pendant l'application des correctifs.

Seuils de simultanéité et d'erreur pour l'option « Corriger maintenant »

Pour les opérations Corriger maintenant, les options de simultanéité et de seuil d'erreur sont gérées par Patch Manager. Il n'est pas nécessaire de spécifier le nombre de nœuds gérés à corriger simultanément, ni le nombre d'erreurs autorisées avant que l'opération échoue. Patch Manager applique les paramètres de simultanéité et de seuil d'erreur décrits dans les tableaux suivants lorsque vous appliquez des correctifs à la demande.

> **Important**
>
> Les seuils suivants s'appliquent aux opérations `Scan and install` uniquement. Pour les opérations `Scan`, Patch Manager tente d'analyser jusqu'à 1 000 nœuds simultanément et de poursuivre l'analyse jusqu'à ce qu'il ait rencontré jusqu'à 1 000 erreurs.

Concurrences : opérations d'installation

Nombre total de nœuds gérés associés à l'opération Patch now (Appliquer les correctifs maintenant)	Nombre de nœuds gérés analysés ou corrigés simultanément
Moins de 25	1
25 à 100	5 %
101 à 1 000	8 %
Plus de 1 000	10 %

Seuil d'erreur : opérations d'installation

Nombre total de nœuds gérés associés à l'opération Patch now (Appliquer les correctifs maintenant)	Nombre d'erreurs autorisées avant que l'opération échoue
Moins de 25	1
25 à 100	5

Nombre total de nœuds gérés associés à l'opération Patch now (Appliquer les correctifs maintenant)	Nombre d'erreurs autorisées avant que l'opération échoue
101 à 1 000	10
Plus de 1 000	10

Utilisation des hooks de cycle de vie « Corriger maintenant »

L'opération Corriger maintenant vous permet d'exécuter des documents SSM Command en tant que hooks de cycle de vie durant une opération d'application de correctifs `Install`. Vous pouvez utiliser ces hooks pour des tâches telles que l'arrêt des applications avant l'application de correctifs ou l'exécution de surveillances de l'état de vos applications après l'application de correctifs ou après un redémarrage.

Pour plus d'informations sur l'utilisation des hooks de cycle de vie, consultez À propos du document SSM `AWS-RunPatchBaselineWithHooks`.

Le tableau suivant répertorie les hooks de cycle de vie disponibles pour chacun des trois options Corriger maintenant, ainsi que des exemples d'utilisation pour chaque hook.

Hooks de cycle de vie et exemples d'utilisation

Option de redémarrage	Hook : avant l'installation	Hook : après l'installation	Hook : à la sortie	Hook : après le redémarrage planifié
Redémarrer si nécessaire	Exécutez un document SSM avant le début de l'application des correctifs. Exemple d'utilisation : arrêtez les applications en toute sécurité avant le début	Exécutez un document SSM à la fin de l'opération d'application des correctifs et avant le redémarrage du nœud géré. Exemple d'utilisation : exécutez	Exécutez un document SSM une fois l'opération de correctif terminée et les instances redémarrées. Exemple d'utilisation : vérifiez que les applicati	Non disponible

Option de redémarrage	Hook : avant l'installation	Hook : après l'installation	Hook : à la sortie	Hook : après le redémarrage planifié
	du processus d'application des correctifs.	des opérations telles que l'installation d'applications tierces avant un redémarrage potentiel.	ons s'exécutent comme prévu après l'application des correctifs.	
Ne pas redémarrer mes instances	Idem ci-dessus.	Exécutez un document SSM à la fin de l'opération d'application des correctifs. Exemple d'utilisation : vérifiez que les applications s'exécutent comme prévu après l'application des correctifs.	Non disponible	Non disponible

Option de redémarrage	Hook : avant l'installation	Hook : après l'installation	Hook : à la sortie	Hook : après le redémarrage planifié
Planifier une heure de redémarrage	Idem ci-dessus.	Identique à Ne pas redémarrer mes instances.	Non disponible	Exécutez un document SSM immédiatement après la fin d'un redémarrage planifié. Exemple d'utilisation : vérifiez que les applications s'exécutent comme prévu après le redémarrage.

Exécution de l'option « Corriger maintenant »

Procédez comme suit pour appliquer des correctifs à la demande à vos nœuds gérés.

Pour exécuter l'option « Corriger maintenant »

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Patch Manager.
3. Sur la AWS Systems Manager Patch Managerpage ou sur la page des lignes de base des correctifs, selon celle qui s'ouvre, choisissez Patch now.
4. Pour Opération d'application des correctifs, sélectionnez l'une des options suivantes :

   - Scan (Analyser) : Patch Manager recherche les correctifs manquants sur vos nœuds gérés, mais ne les installe pas. Vous pouvez afficher les résultats dans le tableau de bord Compliance ou dans les autres outils que vous utilisez pour afficher la conformité des correctifs.

- Scan and install (Analyser et installer) : Patch Manager recherche les correctifs manquants sur vos nœuds gérés et les installe.

5. Effectuez cette étape uniquement si vous avez choisi Analyser et installer à l'étape précédente. Pour Reboot option (Option de redémarrage), sélectionnez l'une des options suivantes :

   - Reboot if needed (Redémarrer si nécessaire) : après l'installation, Patch Manager ne redémarre les nœuds gérés que si cela est nécessaire pour finaliser l'installation des correctifs.
   - Don't reboot my instances (Ne pas redémarrer mes instances) : après l'installation, Patch Manager ne redémarre pas les nœuds gérés. Vous pouvez redémarrer les nœuds manuellement lorsque vous sélectionnez ou gérez les redémarrages en dehors de Patch Manager.
   - Schedule a reboot time (Planifier une heure de redémarrage) : spécifiez la date, l'heure et le fuseau horaire UTC auxquels vous souhaitez que Patch Manager redémarre vos nœuds gérés. Après avoir exécuté l'option Corriger maintenant, le redémarrage planifié est répertorié comme une association dans State Manager sous le nom `AWS-PatchRebootAssociation`.

6. Pour Instances to patch (Instances à corriger), sélectionnez l'une des options suivantes :

   - Corrigez toutes les instances : Patch Manager exécute actuellement l'opération spécifiée sur tous les nœuds gérés Compte AWS de votre instance Région AWS.
   - Patch only the target instances I specify (N'appliquer les correctifs que sur les instances cibles que je spécifie) : vous spécifiez les nœuds gérés à cibler à l'étape suivante.

7. Utilisez cette étape uniquement si vous avez choisi Corriger seulement les instances cibles que je spécifie à l'étape précédente. Dans la section Target selection (Sélection de la cible), identifiez les nœuds sur lesquels vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant les nœuds manuellement ou en spécifiant un groupe de ressources.

   > **Note**
   >
   > Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés 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 étiquetés avec la `aws:cloudformation:stack-id` balise par défaut. Si elle a été supprimée,

cela peut empêcher Patch Manager de déterminer quels nœuds gérés appartiennent au groupe de ressources.

8. (Facultatif) Pour Stockage des journaux d'application des correctifs, si vous voulez créer et enregistrer des journaux à partir de cette opération d'application de correctifs, sélectionnez le compartiment S3 dans lequel les journaux seront stockés.

> **Note**
>
> Les autorisations S3 qui accordent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance (pour les instances EC2) ou de la fonction du service IAM (pour les machines activées par un système hybride) attribués à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer le rôle de service IAM requis pour Systems Manager dans les environnements hybrides et multicloud. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, assurez-vous que le profil d'instance ou le rôle de service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

9. (Facultatif) Pour exécuter des documents SSM en tant que hooks de cycle de vie au niveau de points spécifiques de l'opération d'application de correctifs, procédez comme suit :

   - Sélectionnez Utiliser des hooks de cycle de vie.
   - Pour chaque hook disponible, sélectionnez le document SSM à exécuter au point spécifié de l'opération :
     - Avant l'installation
     - Après l'installation
     - À la sortie
     - Après le redémarrage planifié

   > **Note**
   >
   > Le document par défaut, `AWS-Noop`, n'exécute aucune opération.

10. Sélectionnez Corriger maintenant.

La page Association execution summary (Résumé d'exécution de l'association) s'ouvre. (Le patch utilise désormais des associations dansState Manager, une fonctionnalité de AWS Systems Manager, pour ses opérations.) Dans la zone Operation summary (Résumé de l'opération), vous pouvez surveiller le statut de l'analyse ou de l'application des correctifs sur les nœuds gérés que vous avez spécifiés.

Utilisation des référentiels de correctifs

Un référentiel de correctifs de la fonctionnalité Patch Manager d'AWS Systems Manager définit les correctifs qui peuvent être installés sur vos nœuds gérés. 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 n'est installé sur un nœud géré qu'à condition qu'il soit compatible avec le logiciel du nœud, même si le correctif a par ailleurs été approuvé pour le nœud géré.

Rubriques

- Affichage des référentiels de correctifs prédéfinis AWS
- Utilisation des référentiels de correctifs personnalisés
- Définition d'un référentiel de correctifs existante en tant que valeur par défaut

Plus d'informations

- À propos des références de correctifs

Affichage des référentiels de correctifs prédéfinis AWS

Patch Manager, une fonctionnalité de AWS Systems Manager, inclut une ligne de base de correctifs prédéfinie pour chaque système d'exploitation pris en charge parPatch Manager. Vous pouvez soit utiliser ces références de correctifs (impossibles à personnaliser), soit en créer. La procédure suivante décrit comment afficher un référentiel 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.

Pour afficher les lignes de base de correctifs AWS prédéfinies

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Patch Manager.
3. Dans la liste des références de correctifs, sélectionnez l'ID de l'une des références de correctifs prédéfinies.

   -ou-

   Si vous accédez à Patch Manager pour la première fois dans l' Région AWS actuelle, choisissez Commencer par une présentation, sélectionnez l'onglet Référentiels de correctifs, puis choisissez l'ID de référentiel de l'un des référentiels de correctifs prédéfinis.

   > **Note**
   >
   > Pour Windows Server, trois référentiels de correctifs prédéfinis sont fournis. Les référentiels de correctifs `AWS-DefaultPatchBaseline` et `AWS-WindowsPredefinedPatchBaseline-OS` prennent uniquement en charge les mises à jour du système d'exploitation Windows. `AWS-DefaultPatchBaseline` est utilisé comme référentiel de correctifs par défaut pour les nœuds gérés Windows Server, sauf si vous en spécifiez un autre. Ces deux référentiels de correctifs ont des paramètres de configuration identiques. Le plus récent des deux, `AWS-WindowsPredefinedPatchBaseline-OS`, a été créé pour le différencier du troisième référentiel de correctifs prédéfini pour Windows Server. Ce référentiel de correctifs, `AWS-WindowsPredefinedPatchBaseline-OS-Applications`, peut être utilisé pour appliquer des correctifs à la fois au système d'exploitation Windows Server et aux applications prises en charge publiées par Microsoft.
   > Pour plus d'informations, consultez Définition d'un référentiel de correctifs existante en tant que valeur par défaut.

4. Dans la section Règles d'approbation, consultez la configuration des référentiels de correctifs.
5. Si la configuration est acceptable pour vos nœud gérés, vous pouvez passer à la procédure Utilisation des groupes de correctifs.

-ou-

Pour créer votre propre référentiel de correctifs par défaut, passez à la rubrique Utilisation des référentiels de correctifs personnalisés.

Utilisation des référentiels de correctifs personnalisés

Patch Manager, une fonctionnalité de AWS Systems Manager, inclut un référentiel de correctifs prédéfini pour chaque système d'exploitation pris en charge par Patch Manager. 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 votre propre référentiel de correctifs personnalisé. Pour en savoir plus sur les références de correctifs, consultez À propos des références de correctifs prédéfinies et personnalisées.

Rubriques

- Création d'un référentiel de correctifs personnalisé (Linux)
- Créer un référentiel de correctifs personnalisé (macOS)
- Créer un référentiel de correctifs personnalisé (Windows)
- Mise à jour ou suppression d'un référentiel de correctifs personnalisé

Création d'un référentiel de correctifs personnalisé (Linux)

Utilisez la procédure suivante pour créer une ligne de base de correctifs personnalisée pour les nœuds gérés par Linux dansPatch Manager, une fonctionnalité de AWS Systems Manager.

Pour plus d'informations sur la création d'un référentiel de correctifs pour les nœuds gérés macOS, consultez Créer un référentiel de correctifs personnalisé (macOS). Pour plus d'informations sur la création d'un référentiel de correctifs pour les nœuds gérés Windows, consultez Créer un référentiel de correctifs personnalisé (Windows).

Pour créer un référentiel de correctifs personnalisé pour les nœuds gérés Linux

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Patch Manager.
3. Choisissez l'onglet Référentiels de correctifs, puis choisissez Créer un référentiel de correctifs.

-ou-

Si vous accédez à Patch Manager pour la première fois dans la Région AWS actuelle, choisissez Commencer par une présentation, sélectionnez l'onglet Référentiels de correctifs, puis choisissez Créer un référentiel de correctifs.

4. Pour Nom, entrez un nom pour votre nouvelle référentiel de correctifs, par exemple : `MyRHELPatchBaseline`.
5. (Facultatif) Pour Description, saisissez une description pour cette référence de correctif.
6. Pour Operating system (Système d'exploitation), sélectionnez un système d'exploitation, par exemple, `Red Hat Enterprise Linux`.
7. Si vous souhaitez commencer à utiliser cette référence de correctif comme référence par défaut pour le système d'exploitation sélectionné dès sa création, cochez la case Définir cette référence comme référence de correctif par défaut pour les instances du ***`nom du système d'exploitation`***.

   > **Note**
   >
   > Cette option n'est disponible que si vous avez accédé à Patch Manager pour la première fois avant la publication des politiques de correctifs le 22 décembre 2022.
   > Pour de plus amples informations, sur la définition d'un référentiel de correctifs existante en tant que référence par défaut, veuillez consulter Définition d'un référentiel de correctifs existante en tant que valeur par défaut.

8. Dans la section Règles d'approbation pour les systèmes d'exploitation), utilisez les champs pour créer une ou plusieurs règles d'approbation automatique.

   - Produits : 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 un référentiel de correctifs pour contrôler si des mises à niveau mineures pour Linux sont installées, par exemple RHEL 7.8. Les mises à

niveau mineures de version peuvent être installées automatiquement par Patch Manager, à 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 un référentiel de correctifs les installe.

- Option 1 : La règle d'approbation la plus large pour s'assurer que des mises à niveau mineures sont installées lorsqu'elles sont disponibles consiste à définir la valeur de Classification sur `All` (*) et à choisir l'option Inclusion de mises à jour non liées à la sécurité.
- Option 2 : Pour vous assurer que les correctifs d'une version d'un 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`.

  Saisissez `kernel-3.10.0-*.el7.x86_64` dans la liste Approved patches (Correctifs approuvés) de votre référentiel de correctifs pour vous assurer que tous les correctifs, y compris les mises à niveau de versions mineures, sont appliqués à vos nœuds gérés 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 nœuds gérés, y compris les mises à niveau de versions mineures, en utilisant le InstallOverrideListparamètre indiqué dans le `AWS-RunPatchBaseline` document. Pour plus d'informations, consultez À propos du document SSM `AWS-RunPatchBaseline`.

- Severity (Sévérité) : valeur de sévérité des correctifs à laquelle la règle va s'appliquer, par exemple `Critical`. La sélection par défaut est `All`.
- Approbation automatique : méthode de sélection des patchs 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 : pendant lesquels Patch Manager doit attendre la publication ou la dernière mise à jour d'un correctif avant son approbation automatique. Vous pouvez entrer tout nombre entier situé entre zéro (0) et 360. Nous vous recommandons, en règle générale, de ne pas attendre plus de 100 jours.
  - L'approbation des correctifs publiés jusqu'à une date spécifique: date de publication des correctifs pour laquelle Patch Manager applique automatiquement tous les correctifs publiés à cette date ou avant cette date. Par exemple, si vous spécifiez le 7 juillet 2023, aucun correctif publié ou mis à jour le 8 juillet 2023 ou après ne sera installé automatiquement.
- (Facultatif) Rapport de conformité : niveau de sévérité que vous voulez affecter aux correctifs approuvés par la référence, tel que `Critical` ou `High`.

> **Note**
>
> Si vous spécifiez un niveau de rapport de conformité et que l'état des correctifs d'un correctif approuvé est indiqué `Missing`, le niveau de sévérité de conformité global indiqué pour le référentiel de correctifs est le niveau de sévérité que vous avez spécifié.

- Include non-security updates (Inclure les mises à jour non liées à la sécurité) : cochez cette case pour installer les correctifs non liés à la sécurité pour le système d'exploitation Linux disponibles dans le référentiel source, en plus des correctifs de sécurité.

> **Note**
>
> Pour SUSE Linux Enterprise Server (SLES), il n'est pas nécessaire de cocher cette case, car les correctifs destinés à résoudre les problèmes de sécurité et autres sont installés par défaut sur les nœuds gérés SLES. Pour plus d'informations, consultez le contenu relatif à SLES dans Sélection des correctifs de sécurité.

Pour en savoir plus sur l'utilisation des règles d'approbation dans un référentiel de correctifs personnalisée, consultez À propos des références personnalisées.

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 obtenir des informations sur les formats acceptés pour les listes de correctifs approuvés et de correctifs rejetés, veuillez consulter À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés.

   - (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 Inclusion de mises à jour non liées à la sécurité pour que ces correctifs soient également 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 obtenir des informations sur les formats acceptés pour les listes de correctifs approuvés et de correctifs rejetés, veuillez consulter À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés.

    - Pour Rejected patches action (Action pour les correctifs rejetés), sélectionnez l'action que Patch Manager doit effectuer sur les correctifs inclus dans la liste Rejected patches (Correctifs rejetés).

- Allow as dependency (Autoriser en tant que dépendance) : un package de la liste Rejected patches (Correctifs rejetés) est installé uniquement s'il constitue une dépendance d'un autre package. Il est considéré comme conforme à la ligne de base du correctif et son état est indiqué sous la forme InstalledOther. Il s'agit de l'action par défaut si aucune option n'est spécifiée.
- Bloquer : les packages figurant dans la liste des correctifs rejetés, ainsi que les packages qui les incluent en tant que dépendances, ne sont en Patch Manager aucun cas installés. Si un package a été installé avant d'être ajouté à la liste des correctifs rejetés, ou s'il est installé en dehors de cette Patch Manager période, il est considéré comme non conforme à la ligne de base des correctifs et son état est signalé comme suit. InstalledRejected

11. (Facultatif) Si vous souhaitez spécifier des référentiels de correctifs alternatifs pour différentes versions d'un système d'exploitation, telles que AmazonLinux2016.03 et AmazonLinux2017.09, procédez comme suit pour chaque produit dans la section Sources des 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, saisissez la valeur de la configuration du référentiel yum à utiliser dans le format suivant :

    ```
    [main]
    name=MyCustomRepository
    baseurl=https://my-custom-repository
    enabled=1
    ```

    > **Tip**
    >
    > Pour plus d'informations sur les autres options disponibles pour la configuration de votre référentiel yum, reportez-vous à la section dnf.conf(5).

    Sélectionnez 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).

12. (Facultatif) Pour Gérer les balises, appliquez une ou plusieurs paires nom/valeur de clé de balise au référentiel 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 un référentiel de correctifs pour identifier le niveau de sévérité 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. Sélectionnez Créer un référentiel de correctif.

Créer un référentiel de correctifs personnalisé (macOS)

Utilisez la procédure suivante pour créer une ligne de base de correctifs personnalisée pour les nœuds macOS gérés dansPatch Manager, une fonctionnalité de AWS Systems Manager.

Pour plus d'informations sur la création d'un référentiel de correctifs pour les nœuds gérés Windows Server, consultez Créer un référentiel de correctifs personnalisé (Windows). Pour plus d'informations sur la création d'un référentiel de correctifs pour les nœuds gérés Linux, consultez Création d'un référentiel de correctifs personnalisé (Linux).

> **Note**
>
> macOSn'est pas pris en charge dans tous les cas Régions AWS. Pour plus d'informations sur la prise en charge d'Amazon EC2 pourmacOS, consultez les instances Mac Amazon EC2 dans le guide de l'utilisateur Amazon EC2.

Pour créer un référentiel de correctifs personnalisé pour les nœuds gérés macOS

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Patch Manager.
3. Choisissez l'onglet Référentiels de correctifs, puis choisissez Créer un référentiel de correctifs.

-ou-

Si vous accédez à Patch Manager pour la première fois dans la Région AWS actuelle, choisissez Commencer par une présentation, sélectionnez l'onglet Référentiels de correctifs, puis choisissez Créer un référentiel de correctifs.

4. Pour Nom, entrez un nom pour votre nouvelle référentiel de correctifs, par exemple : `MymacOSPatchBaseline`.
5. (Facultatif) Pour Description, saisissez une description pour cette référence de correctif.
6. Pour Système d'exploitation, sélectionnez macOS.
7. Si vous souhaitez commencer à utiliser ce référentiel de correctifs comme référence par défaut pour macOS dès sa création, cochez la case Set this patch baseline as the default patch baseline for macOS instances (Définir cette référence comme référentiel de correctifs par défaut pour les instances du nom du système d'exploitation).

   > **Note**
   >
   > Cette option n'est disponible que si vous avez accédé à Patch Manager pour la première fois avant la publication des politiques de correctifs le 22 décembre 2022.
   > Pour de plus amples informations, sur la définition d'un référentiel de correctifs existante en tant que référence par défaut, veuillez consulter Définition d'un référentiel de correctifs existante en tant que valeur par défaut.

8. Dans la section Règles d'approbation pour les systèmes d'exploitation), utilisez les champs pour créer une ou plusieurs règles d'approbation automatique.

   - Produits : version des systèmes d'exploitation à laquelle s'applique la règle d'approbation, par exemple `Mojave10.14.1` ou `Catalina10.15.1`. La sélection par défaut est `All`.

     > **Note**
     >
     > Le système de gestion des packages logiciels open source Homebrew a cessé la prise en charge de macOS 10.14.x (Mojave) et 10.15.x (Catalina). Par conséquent, les opérations d'application de correctifs sur ces versions ne sont actuellement pas prises en charge.

- Classification : le ou les gestionnaires de packages dont vous voulez qu'ils appliquent des packages durant le processus d'application de correctifs. Sélectionnez parmi les éléments suivants :
  - softwareupdate
  - installer (programme d'installation)
  - brew
  - brew cask

  La sélection par défaut est `All`.
- (Facultatif) Rapport de conformité : niveau de sévérité que vous voulez affecter aux correctifs approuvés par la référence, tel que `Critical` ou `High`.

  > **Note**
  >
  > Si vous spécifiez un niveau de rapport de conformité et que l'état des correctifs d'un correctif approuvé est indiqué `Missing`, le niveau de sévérité de conformité global indiqué pour le référentiel de correctifs est le niveau de sévérité que vous avez spécifié.
- Include non-security updates (Inclure les mises à jour non liées à la sécurité) : cochez cette case pour installer les correctifs non liés à la sécurité pour le système d'exploitation disponibles dans le référentiel source, en plus des correctifs de sécurité.

Pour en savoir plus sur l'utilisation des règles d'approbation dans un référentiel de correctifs personnalisée, consultez À propos des références personnalisées.

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 obtenir des informations sur les formats acceptés pour les listes de correctifs approuvés et de correctifs rejetés, veuillez consulter À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés.

- (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 Inclusion de mises à jour non liées à la sécurité pour que ces correctifs soient également installés sur votre système d'exploitation macOS.

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 obtenir des informations sur les formats acceptés pour les listes de correctifs approuvés et de correctifs rejetés, veuillez consulter À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés.

    - Pour Rejected patches action (Action pour les correctifs rejetés), sélectionnez l'action que Patch Manager doit effectuer sur les correctifs inclus dans la liste Rejected patches (Correctifs rejetés).
      - Allow as dependency (Autoriser en tant que dépendance) : un package de la liste Rejected patches (Correctifs rejetés) est installé uniquement s'il constitue une dépendance d'un autre package. Il est considéré comme conforme à la ligne de base du correctif et son état est indiqué sous la forme InstalledOther. Il s'agit de l'action par défaut si aucune option n'est spécifiée.
      - Bloquer : les packages figurant dans la liste des correctifs rejetés et les packages qui les incluent en tant que dépendances ne sont en Patch Manager aucun cas installés. Si un package a été installé avant d'être ajouté à la liste des correctifs rejetés, ou s'il est installé

en dehors de cette Patch Manager période, il est considéré comme non conforme à la ligne de base des correctifs et son état est signalé comme. InstalledRejected

11. (Facultatif) Pour Gérer les balises, appliquez une ou plusieurs paires nom/valeur de clé de balise au référentiel 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 un référentiel de correctifs pour identifier le niveau de sévérité de correctifs qu'elle spécifie, le gestionnaire de packages auquel 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=PackageManager,Value=softwareupdate`
    - `Key=Environment,Value=Production`

12. Sélectionnez Créer un référentiel de correctif.

Créer un référentiel de correctifs personnalisé (Windows)

Utilisez la procédure suivante pour créer une ligne de base de correctifs personnalisée pour les nœuds gérés par Windows dansPatch Manager, une fonctionnalité de AWS Systems Manager.

Pour plus d'informations sur la création d'un référentiel de correctifs pour les nœuds gérés Linux, consultez Création d'un référentiel de correctifs personnalisé (Linux). Pour plus d'informations sur la création d'un référentiel de correctifs pour les nœuds gérés macOS, consultez Créer un référentiel de correctifs personnalisé (macOS).

Pour obtenir un exemple de création d'un référentiel de correctifs limitée à l'installation des Service Packs Windows uniquement, veuillez consulter Didacticiel : créer un référentiel de correctifs pour l'installation des Service Packs Windows (console).

Pour créer un référentiel de correctifs personnalisée (Windows)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Patch Manager.
3. Choisissez l'onglet Référentiels de correctifs, puis choisissez Créer un référentiel de correctifs.

-ou-

Si vous accédez à Patch Manager pour la première fois dans la Région AWS actuelle, choisissez Commencer par une présentation, sélectionnez l'onglet Référentiels de correctifs, puis choisissez Créer un référentiel de correctifs.

4. Pour Nom, entrez un nom pour votre nouvelle référentiel de correctifs, par exemple : `MyWindowsPatchBaseline`.
5. (Facultatif) Pour Description, saisissez une description pour cette référence de correctif.
6. Pour Système d'exploitation, sélectionnez `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.

> **Note**
>
> Cette option n'est disponible que si vous avez accédé à Patch Manager pour la première fois avant la publication des politiques de correctifs le 22 décembre 2022.
> Pour de plus amples informations, sur la définition d'un référentiel de correctifs existante en tant que référence par défaut, veuillez consulter Définition d'un référentiel de correctifs existante en tant que valeur par défaut.

8. Dans la section Règles d'approbation pour les systèmes d'exploitation), utilisez les champs pour créer une ou plusieurs règles d'approbation automatique.

   - Produits : version des systèmes d'exploitation à laquelle s'applique la règle d'approbation, par exemple `WindowsServer2012`. La sélection par défaut est `All`.
   - Classification : type de correctifs auquel s'applique la règle d'approbation, par exemple `CriticalUpdates`, `Drivers` et `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, consultez Didacticiel : créer un référentiel de correctifs pour l'installation des Service Packs Windows (console).

- Severity (Sévérité) : valeur de sévérité des correctifs à laquelle la règle va s'appliquer, par exemple `Critical`. La sélection par défaut est `All`.
- Approbation automatique : méthode de sélection des patchs pour approbation automatique.
  - Approuver les correctifs après un nombre de jours spécifié : pendant lesquels Patch Manager doit attendre après la publication ou la mise à jour d'un correctif avant son approbation automatique. Vous pouvez entrer tout nombre entier situé entre zéro (0) et 360. Nous vous recommandons, en règle générale, de ne pas attendre plus de 100 jours.
  - L'approbation des correctifs publiés jusqu'à une date spécifique: date de publication des correctifs pour laquelle Patch Manager applique automatiquement tous les correctifs publiés à cette date ou avant cette date. Par exemple, si vous spécifiez le 7 juillet 2023, aucun correctif publié ou mis à jour le 8 juillet 2023 ou après ne sera installé automatiquement.
- (Facultatif) Compliance reporting (Rapport de conformité) : niveau de sévérité que vous voulez affecter aux correctifs approuvés par la référence, tel que `High`.

> **Note**
>
> Si vous spécifiez un niveau de rapport de conformité et que l'état des correctifs d'un correctif approuvé est indiqué `Missing`, le niveau de sévérité de conformité global indiqué pour le référentiel de correctifs est le niveau de sévérité que vous avez spécifié.

9. (Facultatif) Dans la section Approval rules for applications (Règles d'approbation pour les applications Microsoft), utilisez les champs pour créer une ou plusieurs règles d'approbation automatique.

> **Note**
>
> Au lieu de spécifier des règles d'approbation, vous pouvez spécifier des listes de correctifs approuvés et de correctifs rejetés en tant qu'exceptions de correctifs. Voir les étapes 10 et 11.

- Product family (Famille de produits) : famille de produits Microsoft générale pour laquelle vous souhaitez spécifier une règle, par exemple `Office` ou `Exchange Server`.

- Produits : version de l'application à laquelle s'applique la règle d'approbation, par exemple `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`.
- Severity (Sévérité) : valeur de sévérité des correctifs à laquelle la règle s'applique, par exemple `Critical`. La sélection par défaut est `All`.
- Approbation automatique : méthode de sélection des patchs pour approbation automatique.
  - Approuver les correctifs après un nombre de jours spécifié : pendant lesquels Patch Manager doit attendre après la publication ou la mise à jour d'un correctif avant son approbation automatique. Vous pouvez entrer tout nombre entier situé entre zéro (0) et 360. Nous vous recommandons, en règle générale, de ne pas attendre plus de 100 jours.
  - L'approbation des correctifs publiés jusqu'à une date spécifique: date de publication des correctifs pour laquelle Patch Manager applique automatiquement tous les correctifs publiés à cette date ou avant cette date. Par exemple, si vous spécifiez le 7 juillet 2023, aucun correctif publié ou mis à jour le 8 juillet 2023 ou après ne sera installé automatiquement.
- (Facultatif) Rapport de conformité : niveau de sévérité que vous voulez affecter aux correctifs approuvés par la référence, tel que `Critical` ou `High`.

> **Note**
>
> Si vous spécifiez un niveau de rapport de conformité et que l'état des correctifs d'un correctif approuvé est indiqué `Missing`, le niveau de sévérité de conformité global indiqué pour le référentiel de correctifs est le niveau de sévérité que vous avez spécifié.

10. (Facultatif) Si, au lieu que des correctifs soient sélectionnés selon les règles d'approbation, vous voulez les approuver explicitement, procédez comme suit dans la section Exceptions de correctifs :

    - Pour Approved patches (Correctifs approuvés), entrez une liste séparée par des virgules des correctifs à approuver.

> **Note**
>
> Pour obtenir des informations sur les formats acceptés pour les listes de correctifs approuvés et de correctifs rejetés, veuillez consulter À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés.

- (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 obtenir des informations sur les formats acceptés pour les listes de correctifs approuvés et de correctifs rejetés, veuillez consulter À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés.

    - Pour Rejected patches action (Action pour les correctifs rejetés), sélectionnez l'action que Patch Manager doit effectuer sur les correctifs inclus dans la liste Rejected patches (Correctifs rejetés).
      - Allow as dependency (Autoriser en tant que dépendance) : un package de la liste Rejected patches (Correctifs rejetés) est installé uniquement s'il constitue une dépendance d'un autre package. Il est considéré comme conforme à la ligne de base du correctif et son état est indiqué sous la forme InstalledOther. Il s'agit de l'action par défaut si aucune option n'est spécifiée.
      - Bloquer : les packages figurant dans la liste des correctifs rejetés, ainsi que les packages qui les incluent en tant que dépendances, ne sont en Patch Manager aucun cas installés. Si un package a été installé avant d'être ajouté à la liste des correctifs rejetés, ou s'il est installé en dehors de cette Patch Manager période, il est considéré comme non conforme à la ligne de base des correctifs et son état est signalé comme suit. InstalledRejected

12. (Facultatif) Pour Gérer les balises, appliquez une ou plusieurs paires nom/valeur de clé de balise au référentiel 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 un référentiel de correctifs pour identifier le niveau de sévérité 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. Sélectionnez Créer un référentiel de correctif.

Mise à jour ou suppression d'un référentiel de correctifs personnalisé

Vous pouvez mettre à jour ou supprimer une ligne de base de correctifs personnalisée que vous avez créée dansPatch Manager, une fonctionnalité de AWS Systems Manager. Lorsque vous mettez à jour un référentiel 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 au référentiel de correctifs. Vous ne pouvez pas modifier le type de système d'exploitation pour lequel une ligne de base de correctifs a été créée, ni apporter de modifications à une ligne de base de correctifs prédéfinie fournie par AWS.

Mise à jour ou suppression d'un référentiel de correctifs

Suivez les étapes ci-dessous pour mettre à jour ou supprimer un référentiel de correctifs.

> **Important**
>
> Faites preuve de vigilance lorsque vous supprimez un référentiel de correctifs personnalisé susceptible d'être utilisé par la configuration d'une politique de correctifs dans Quick Setup. Si vous utilisez une configuration de politique de correctifs dans Quick Setup, les mises à jour que vous apportez aux référentiels de correctifs personnalisés sont synchronisées avec Quick Setup une fois par heure.
>
> Si un référentiel de correctifs personnalisé référencé dans une politique de correctifs est supprimé, une bannière s'affiche sur la page Quick Setup Configuration details (Détails de configuration) de votre politique de correctifs. La bannière vous informe que la politique de correctifs fait référence à un référentiel de correctifs qui n'existe plus et que les opérations

d'application de correctifs suivantes échoueront. Dans ce cas, revenez à la page Quick Setup Configurations, sélectionnez la configuration Patch Manager, puis choisissez Actions, Edit configuration (Modifier la configuration). Le nom du référentiel de correctifs supprimé est surligné et vous devez sélectionner un nouveau référentiel de correctifs pour le système d'exploitation concerné.

Pour mettre à jour ou supprimer un référentiel de correctifs

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Patch Manager.
3. Sélectionnez le référentiel de correctifs à mettre à jour ou supprimer, puis exécutez l'une des tâches suivantes :

   - Pour supprimer la ligne de base des correctifs de votre Compte AWS ordinateur, choisissez Supprimer. Le système vous invite à confirmer vos actions.
   - Pour modifier le nom ou la description d'un référentiel de correctifs, les règles d'approbation ou les exceptions de correctifs, sélectionnez Modifier. Sur la page Modifier le référentiel de correctif, modifiez les valeurs et options souhaitées, puis sélectionnez Enregistrer les modifications.
   - Pour ajouter, modifier ou supprimer des balises appliquées au référentiel de correctifs, sélectionnez l'onglet Balises, puis Modifier les balises. Sur la page Edit patch baseline tags (Modifier les balises de référentiel de correctif), mettez à jour les balises du référentiel de correctifs, puis sélectionnez Enregistrer les modifications.

   Pour de plus amples informations sur les choix de configuration que vous pouvez effectuer, veuillez consulter Utilisation des référentiels de correctifs personnalisés.

Définition d'un référentiel de correctifs existante en tant que valeur par défaut

> **Important**
>
> Les sélections de référentiel de correctifs par défaut que vous effectuez ici ne s'appliquent pas aux opérations d'application de correctifs basées sur une politique de correctifs. Les politiques de correctifs utilisent leurs propres spécifications de référentiel de correctifs. Pour

plus d'informations sur les politiques de correctifs, veuillez consulter la rubrique Utilisation des stratégies de correctifs Quick Setup.

Lorsque vous créez un référentiel de correctifs personnalisée dans Patch Manager, une des fonctionnalités de AWS Systems Manager, vous pouvez la définir comme référence par défaut pour le type de système d'exploitation associé dès sa création. Pour plus d'informations, veuillez consulter Utilisation des référentiels de correctifs personnalisés.

Vous pouvez également définir un référentiel de correctifs existante en tant que référence par défaut pour un type de système d'exploitation.

> **Note**
>
> Les étapes à suivre varient selon si vous avez accédé pour la première fois à Patch Manager avant ou après la publication des politiques de correctifs le 22 décembre 2022. Si vous avez utilisé Patch Manager avant cette date, vous pouvez utiliser la procédure de la console. Dans le cas contraire, suivez la AWS CLI procédure. Le menu Actions référencé dans la procédure de la console ne s'affiche pas dans les régions où Patch Manager n'a pas été utilisé avant la publication des politiques de correctifs.

Pour définir un référentiel de correctifs comme référence par défaut

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Patch Manager.
3. Sélectionnez l'onglet Patch baselines (Référentiels de correctifs).
4. Dans la liste des références de correctifs, sélectionnez le bouton correspondant à un référentiel de correctifs qui n'est pas actuellement définie comme référence par défaut pour un type de système d'exploitation.

   La colonne Référence par défaut indique les références qui sont actuellement définies comme références par défaut.
5. Dans le menu Actions, sélectionnez Définir un référentiel de correctif par défaut.

> **Important**
>
> Le menu Actions n'est pas disponible si vous n'avez pas travaillé avec Patch Manager la version actuelle Compte AWS et la région avant le 22 décembre 2022. Pour plus d'informations, reportez-vous à la note figurant plus haut dans cette rubrique.

6. Dans la boîte de dialogue de confirmation, sélectionnez Set default (Définir par défaut).

Pour définir un référentiel de correctifs comme référence par défaut (AWS CLI)

1. Exécutez la commande describe-patch-baselines pour afficher la liste des référentiels de correctifs disponibles, ainsi que leurs ID et Amazon Resource Names (ARN).

```
aws ssm describe-patch-baselines
```

2. Exécutez la commande register-default-patch-baseline pour définir un référentiel par défaut pour le système d'exploitation auquel il est associé. Remplacez *baseline-id-or-ARN* par l'ID de la ligne de base de correctif personnalisée ou de la ligne de base prédéfinie à utiliser.

   Linux & macOS

   ```
   aws ssm register-default-patch-baseline \
       --baseline-id baseline-id-or-ARN
   ```

   Vous trouverez ci-dessous un exemple de définition d'un référentiel personnalisé comme référentiel par défaut.

   ```
   aws ssm register-default-patch-baseline \
       --baseline-id pb-abc123cf9bEXAMPLE
   ```

   Voici un exemple de définition d'une ligne de base prédéfinie gérée par AWS défaut.

   ```
   aws ssm register-default-patch-baseline \
       --baseline-id arn:aws:ssm:us-east-2:733109147000:patchbaseline/
   pb-0574b43a65ea646e
   ```

Windows Server

```
aws ssm register-default-patch-baseline ^
    --baseline-id baseline-id-or-ARN
```

Vous trouverez ci-dessous un exemple de définition d'un référentiel personnalisé comme référentiel par défaut.

```
aws ssm register-default-patch-baseline ^
    --baseline-id pb-abc123cf9bEXAMPLE
```

Voici un exemple de définition d'une ligne de base prédéfinie gérée par AWS défaut.

```
aws ssm register-default-patch-baseline ^
    --baseline-id arn:aws:ssm:us-east-2:733109147000:patchbaseline/
pb-071da192df1226b63
```

Affichage des correctifs disponibles

Avec Patch Manager une fonctionnalité de AWS Systems Manager, vous pouvez afficher tous les correctifs disponibles pour un système d'exploitation spécifique et, éventuellement, une version spécifique du système d'exploitation.

> **Tip**
>
> Pour générer une liste des correctifs disponibles et les enregistrer dans un fichier, vous pouvez utiliser la commande describe-available-patches et spécifier votre sortie.

Pour afficher les correctifs disponibles

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Patch Manager.
3. Sélectionnez l'onglet Patches (Correctifs).

   -ou-

Si vous accédez à Patch Manager pour la première fois dans la Région AWS actuelle, choisissez Commencer par une présentation, puis sélectionnez l'onglet Correctifs.

> **Note**
>
> Pour Windows Server, l'onglet Correctifs affiche les mises à jour disponibles auprès de Windows Server Update Service (WSUS).

4. Pour Système d'exploitation, sélectionnez le système d'exploitation pour lequel vous voulez afficher les correctifs disponibles, `Windows` ou `Amazon Linux` par exemple.
5. (Facultatif) Pour Produit, sélectionnez une version du système d'exploitation, `WindowsServer2019` ou `AmazonLinux2018.03` par exemple.
6. (Facultatif) Pour ajouter ou supprimer des colonnes d'informations pour vos résultats, sélectionnez le bouton () en haut à droite de la liste Correctifs. (Par défaut, l'onglet Correctifs affiche des colonnes pour quelques-unes des métadonnées de correctif disponibles seulement.)

   Pour obtenir des informations sur les types de métadonnées qui peuvent être ajoutées à votre affichage, veuillez consulter Correctif dans la Référence des API AWS Systems Manager .

Utilisation des groupes de correctifs

Si vous n'utilisez pas de stratégies de correction dans vos opérations, vous pouvez organiser vos efforts de correction en ajoutant des nœuds gérés à des groupes de correctifs à l'aide de balises.

> **Important**
>
> Les groupes de correctifs ne sont pas utilisés dans les opérations d'application de correctifs basées sur des politiques de correctifs. Pour plus d'informations sur l'utilisation des politiques de correctifs, consultez la rubrique Utilisation des stratégies de correctifs Quick Setup.

Pour utiliser les balises dans les opérations de correctif, vous devez appliquer la clé de balise `Patch Group` ou `PatchGroup` à vos nœuds gérés. Vous devez également spécifier le nom que vous voulez

donner au groupe de correctifs comme valeur de la balise. Vous pouvez spécifier n'importe quelle valeur de balise, mais la clé de balise doit être `Patch Group` ou `PatchGroup`.

`PatchGroup` (sans espace) est nécessaire si vous avez autorisé les balises dans les métadonnées d'instance EC2..

Après avoir regroupé vos nœuds gérés à l'aide de balises, vous devez ajouter la valeur du groupe de correctifs à un référentiel de correctifs. En enregistrant le groupe de correctifs auprès d'un référentiel de correctifs, vous veillez à ce que les correctifs appropriés soient installés lors de l'opération d'application des correctifs. Pour de plus amples informations sur les groupes de correctifs, consultez À propos des groupes de correctifs.

Effectuez les tâches de cette rubrique pour préparer vos nœuds gérés à l'application de correctifs à l'aide de balises avec vos nœuds et votre référentiel de correctifs. La tâche 1 n'est requise que si vous corrigez des instances Amazon EC2. La tâche 2 est requise uniquement si vous corrigez des instances non-EC2 dans un environnement hybride et multicloud. La tâche 3 est requise pour tous les nœuds gérés.

> **Tip**
>
> Vous pouvez également ajouter des balises aux nœuds gérés à l'aide de la AWS CLI commande `add-tags-to-resource` ou de l'opération de l'API Systems Manager`AddTagsToResource`.

Tâches

- Tâche 1 : Ajout d'instances EC2 à un groupe de correctifs à l'aide de balises
- Tâche 2 : Ajout de nœuds gérés à un groupe de correctifs à l'aide de balises
- Tâche 3 : Ajout d'un groupe de correctifs à un référentiel de correctifs

Tâche 1 : Ajout d'instances EC2 à un groupe de correctifs à l'aide de balises

Vous pouvez ajouter des balises aux instances EC2 à l'aide de la console Systems Manager ou de la console Amazon EC2. Cette tâche n'est requise que si vous corrigez des instances Amazon EC2.

> **Important**
>
> Vous ne pouvez pas appliquer la `Patch Group` balise (avec un espace) à une instance Amazon EC2 si l'option Autoriser les balises dans les métadonnées d'instance est activée sur celle-ci. L'autorisation des identifications dans les métadonnées d'instance empêche les noms de clés d'identification de contenir des espaces. Si vous avez autorisé les balises dans les métadonnées des instances EC2, vous devez utiliser la clé de la balise `PatchGroup` (sans espace).

Option 1 : pour ajouter des instances EC2 à un groupe de correctifs (console Systems Manager)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Dans la liste Nœuds gérés, sélectionnez l'ID d'une instance EC2 gérée que vous souhaitez configurer pour l'application de correctifs. Les ID de nœud pour les instances EC2 commencent par `i-`.

   > **Note**
   >
   > Lorsque vous utilisez la console Amazon EC2 AWS CLI, il est possible d'appliquer des `Key = PatchGroup` balises à `Key = Patch Group` des instances qui ne sont pas encore configurées pour être utilisées avec Systems Manager.
   > Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés pour obtenir des conseils de dépannage.

4. Choisissez l'onglet Balises, puis Modifier.
5. Dans la colonne de gauche, saisissez **`Patch Group`** ou **`PatchGroup`**. Si vous avez autorisé les balises dans les métadonnées des instances EC2, il est impératif d'utiliser `PatchGroup` (sans espace).
6. Dans la colonne de droite, saisissez une valeur de balise qui servira de nom au groupe de correctifs.
7. Choisissez Enregistrer.
8. Répétez cette procédure pour ajouter d'autres instances EC2 au même groupe de correctifs.

Option 2 : pour ajouter des instances EC2 à un groupe de correctifs (console Amazon EC2)

1. Ouvrez la console Amazon EC2, puis sélectionnez Instances dans le panneau de navigation.
2. Dans la liste d'instances, sélectionnez une instance que vous souhaitez configurer pour l'application de correctifs.
3. Dans le menu Actions, sélectionnez Paramètres de l'instance, Gérer les balises.
4. Sélectionnez Ajouter une nouvelle balise.
5. Pour Key (Clé), saisissez **`Patch Group`** ou **`PatchGroup`**. Si vous avez autorisé les balises dans les métadonnées des instances EC2, il est impératif d'utiliser `PatchGroup` (sans espace).
6. Pour Valeur, entrez une valeur qui servira de nom au groupe de correctifs.
7. Sélectionnez Enregistrer.
8. Répétez cette procédure pour ajouter d'autres instances au même groupe de correctifs.

Tâche 2 : Ajout de nœuds gérés à un groupe de correctifs à l'aide de balises

Suivez les étapes décrites dans cette rubrique pour ajouter des balises aux appareils AWS IoT Greengrass principaux et aux nœuds gérés non activés par un système hybride EC2 (mi-*). Cette tâche n'est requise que si vous corrigez des instances non-EC2 dans un environnement hybride et multicloud.

> **Note**
>
> Vous ne pouvez pas ajouter de balises sur des nœuds gérés non EC2 via la console Amazon EC2.

Pour ajouter des nœuds gérés non EC2 à un groupe de correctifs (console Systems Manager)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Dans la liste Nœuds gérés, sélectionnez le nom du nœud géré que vous souhaitez configurer pour l'application de correctifs.

> **Note**
>
> Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés pour obtenir des conseils de dépannage.

4. Choisissez l'onglet Balises, puis Modifier.
5. Dans la colonne de gauche, saisissez **`Patch Group`** ou **`PatchGroup`**. Si vous avez autorisé les balises dans les métadonnées des instances EC2, il est impératif d'utiliser `PatchGroup` (sans espace).
6. Dans la colonne de droite, saisissez une valeur de balise qui servira de nom au groupe de correctifs.
7. Sélectionnez Enregistrer.
8. Répétez cette procédure pour ajouter d'autres nœuds gérés dans le même groupe de correctifs.

Tâche 3 : Ajout d'un groupe de correctifs à un référentiel de correctifs

Pour associer un référentiel de correctifs spécifique à vos nœuds gérés, vous devez ajouter la valeur du groupe de correctifs à ce référentiel. En enregistrant le groupe de correctifs auprès d'un référentiel de correctifs, vous veillez à ce que les correctifs appropriés soient installés lors de l'exécution de l'application des correctifs. Cette tâche est requise, que vous corrigiez des instances EC2, des nœuds gérés non EC2 ou les deux.

Pour de plus amples informations sur les groupes de correctifs, consultez À propos des groupes de correctifs.

> **Note**
>
> Les étapes à suivre varient selon si vous avez accédé pour la première fois à Patch Manager avant ou après la publication des stratégies de correctifs le 22 décembre 2022.

Pour ajouter un groupe de correctifs à un référentiel de correctifs (console Systems Manager)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.

2. Dans le panneau de navigation, sélectionnez Patch Manager.
3. Si vous accédez à Patch Manager pour la première fois dans l' Région AWS actuelle et que la page de démarrage de Patch Manager s'ouvre, choisissez Commencer par une présentation.
4. Sélectionnez l'onglet Référentiel de correctifs, puis dans la liste Référentiel de correctifs, sélectionnez le nom du référentiel de correctifs que vous souhaitez configurer pour votre groupe de correctifs.

   Si vous n'avez accédé pour la première fois à Patch Manager qu'après la publication des stratégies de correctifs, vous devez choisir un référentiel personnalisé que vous avez créé.
5. Si la page de détails ID de référence comprend un menu Actions procédez comme suit :

   - Sélectionnez Actions, puis Modifier les groupes de correctifs.
   - Saisissez la valeur de balise que vous avez ajoutée à vos nœuds gérés dans Tâche 2 : Ajout de nœuds gérés à un groupe de correctifs à l'aide de balises, puis sélectionnez Ajouter.

   Si la page de détails ID de référence ne comporte pas un menu Actions les groupes de correctifs ne peuvent pas être configurés dans la console. Au lieu de cela, vous pouvez effectuer l'une des actions suivantes :

   - (Recommandé) Configurez une politique de correctifsQuick Setup, une fonctionnalité de AWS Systems Manager, pour mapper une ligne de base de correctifs à une ou plusieurs instances EC2.

     Pour en savoir plus, consultez les rubriques Utilisation des politiques de correctifs de Quick Setup et Automatiser les correctifs à l'échelle de l'organisation à l'aide d'une politique de correctifs de Quick Setup.
   - Utilisez la register-patch-baseline-for-patch-groupcommande du AWS Command Line Interface (AWS CLI) pour configurer un groupe de correctifs.

Utilisation des paramètres Patch Manager

Rubriques

- Intégration Patch Manager avec AWS Security Hub

Intégration Patch Manager avec AWS Security Hub

AWS Security Hubvous fournit une vue complète de votre état de sécurité dans AWS. Security Hub collecte des données de sécurité provenant de l'ensemble Comptes AWS des Services AWS produits partenaires et pris en charge par des tiers. Avec Security Hub, vous pouvez vérifier votre environnement par rapport aux normes et aux meilleures pratiques de l'industrie de la sécurité. Security Hub vous aide à analyser les tendances en matière de sécurité et à identifier les problèmes de sécurité hautement prioritaires.

En utilisant l'intégration entre Security HubPatch Manager, une fonctionnalité de AWS Systems Manager et Security Hub, vous pouvez envoyer des informations concernant des nœuds non conformes depuis Patch Manager Security Hub. Vous pouvez observer parmi les résultats l'enregistrement d'une vérification de sécurité ou d'une détection liée à la sécurité. Security Hub peut ensuite inclure ces résultats liés aux correctifs dans son analyse de votre posture de sécurité.

Les informations des rubriques suivantes s'appliquent, quels que soient la méthode ou le type de configuration que vous utilisez pour vos opérations d'application de correctifs :

- Une politique de correctifs configurée dans Quick Setup
- Une option de gestion des hôtes configurée dans Quick Setup
- Une fenêtre de maintenance pour exécuter un correctif `Scan` ou une tâche `Install`
- Une opération Patch now (Appliquer les correctifs maintenant) à la demande

Table des matières

Comment Patch Manager envoie des résultats à Security Hub

Dans Security Hub, les problèmes de sécurité sont suivis en tant que findings. (résultats) Certains résultats proviennent de problèmes détectés par d'autres partenaires Services AWS ou par des partenaires tiers. Security Hub utilise également un ensemble de règles pour détecter les problèmes de sécurité et générer des résultats.

Patch Manager est une des fonctionnalités du Systems Manager qui envoie les résultats à Security Hub. Après avoir effectué une opération de correction en exécutant un document SSM (`AWS-RunPatchBaseline`,, ou`AWS-RunPatchBaselineWithHooks`)`AWS-RunPatchBaselineAssociation`, les informations d'application des correctifs sont envoyées à Inventory ou Compliance, aux fonctionnalités de AWS Systems Manager, ou aux deux. Après qu'Inventory, Compliance, ou les deux, ont reçu les données, Patch Manager reçoit une notification. Ensuite, Patch Manager évalue l'exactitude, le formatage et la conformité des données. Si toutes les conditions sont remplies, Patch Manager transmet les données à Security Hub.

Security Hub fournit des outils permettant de gérer les résultats provenant de toutes ces sources. Vous pouvez afficher et filtrer les listes de résultats et afficher les informations sur un résultat. Pour de plus amples informations, consultez la section Viewing findings (Affichage des résultats) dans le Guide de l'utilisateur AWS Security Hub . Vous pouvez également suivre le statut d'une analyse dans un résultat. Pour de plus amples informations, veuillez consulter Prendre des mesure en fonction des résultats dans le Guide de l'utilisateur AWS Security Hub .

Tous les résultats de Security Hub utilisent un format JSON standard appelé AWS Security Finding Format (ASFF). Le format ASFF comprend des informations sur la source du problème, les ressources affectées et le statut actuel du résultat. Pour de plus amples informations, veuillez consulter AWS Security Finding Format (ASFF) dans le Guide de l'utilisateur AWS Security Hub .

Types de résultats que Patch Manager envoie

Patch Manager envoie les résultats à Security Hub dans le format AWS Security Finding Format (ASFF). Dans le format ASFF, le champ `Types` fournit le type de résultat. Les résultats de Patch Manager peuvent avoir la valeur suivante pour `Types` :

- Vérifications de logiciels et de configuration/Gestion des correctifs

Patch Manager envoie un résultat par nœud géré non conforme. Le résultat est signalé avec le type de ressource `AwsEc2Instance` pour pouvoir le mettre ne corrélation avec d'autres intégrations Security Hub qui signalent des types de ressources `AwsEc2Instance`. Patch Manager ne transmet

un résultat à Security Hub que si l'opération a découvert que le nœud géré n'était pas conforme. Le résultat contient les résultats du Résumé des correctifs.

> **Note**
>
> Après avoir signalé un nœud non conforme à Security Hub. Patch Managern'envoie pas de mise à jour à Security Hub une fois que le nœud est conforme. Vous pouvez résoudre manuellement les résultats dans Security Hub une fois que les correctifs requis ont été appliqués au nœud géré.

Pour plus d'informations sur les définitions de conformité, consultez Comprendre les valeurs d'état de conformité des correctifs. Pour plus d'informations à ce sujet`PatchSummary`, consultez PatchSummaryla référence de AWS Security Hub l'API.

Latence pour l'envoi des résultats

Quand Patch Manager crée un résultat, ce dernier est généralement envoyé à Security Hub dans un délai de quelques secondes à 2 heures. La vitesse dépend du trafic en cours de traitement dans la Région AWS à ce moment-là.

Réessayer lorsque Security Hub n'est pas disponible

En cas de panne de service, une AWS Lambda fonction est exécutée pour remettre les messages dans la file d'attente principale après la réexécution du service. Une fois les messages dans la file d'attente principale, la nouvelle tentative est automatique.

Si Security Hub n'est pas disponible, Patch Manager essaie de renvoyer les résultats jusqu'à ce qu'ils soient reçus.

Afficher les résultats dans Security Hub

Cette procédure explique comment consulter les résultats dans Security Hub concernant les nœuds gérés de votre flotte qui ne sont pas conformes aux correctifs.

Pour examiner les résultats de Security Hub en matière de conformité aux correctifs

1. Connectez-vous à la AWS Security Hub console AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/securityhub/.
2. Dans le volet de navigation, choisissez Conclusions.

3. Choisissez la case Ajouter des filtres
   ( ).
4. Dans le menu, sous Filtres, choisissez Nom du produit.
5. Dans la boîte de dialogue qui s'ouvre, sélectionnez est dans le premier champ, puis saisissez **`Systems Manager Patch Manager`** dans le deuxième champ.
6. Choisissez Appliquer.
7. Ajoutez les filtres supplémentaires que vous souhaitez pour affiner vos résultats.
8. Dans la liste des résultats, choisissez le titre d'un résultat sur lequel vous souhaitez obtenir plus d'informations.

   Un volet s'ouvre sur le côté droit de l'écran. Il contient plus de détails sur la ressource, le problème découvert et les solutions recommandées.

   > **Important**
   >
   > À l'heure actuelle, Security Hub indique le type de ressource de tous les nœuds gérés sous la forme `EC2 Instance`. Cela inclut les serveurs sur site et les machines virtuelles (VM) que vous avez enregistrés pour une utilisation avec Systems Manager.

Classifications de sévérité

La liste des résultats de **`Systems Manager Patch Manager`** comprend un rapport sur la sévérité du résultat. Les niveaux de sévérité sont les suivants, du plus faible au plus élevé :

- INFORMATIF : aucun problème n'a été détecté.
- FAIBLE : le problème ne nécessite aucune correction.
- MOYEN : le problème doit être traité, mais n'est pas urgent.
- ÉLEVÉ : le problème doit être traité en priorité.
- CRITIQUE : le problème doit être résolu immédiatement pour éviter qu'il ne s'aggrave.

La sévérité est déterminée par le package non conforme le plus sévère d'une instance. Comme vous pouvez disposer de plusieurs référentiels de correctifs avec différents niveaux de sévérité, le niveau de sévérité le plus élevé est indiqué parmi tous les packages non conformes. Supposons, par exemple, que vous ayez deux packages non conformes. Le niveau de sévérité du package A est « critique » et celui du package B est « faible ». La sévérité sera signalée comme étant « critique ».

Veuillez noter que le champ de sévérité est directement corrélé au champ `Compliance` de Patch Manager. Il s'agit d'un champ que vous attribuez aux correctifs individuels qui correspondent à la règle. Ce champ `Compliance` étant affecté à des correctifs individuels, il n'est pas reflété au niveau du résumé des correctifs.

Contenu connexe

- Résultats du Guide de l'utilisateur AWS Security Hub
- Conformité aux correctifs multicomptes avec Patch Manager et Security Hub sur le Blog AWS de gestion et gouvernance (en anglais)

Résultats types de Patch Manager

Patch Manager envoie les résultats à Security Hub dans le format AWS Security Finding Format (ASFF).

Voici un exemple de résultat type de Patch Manager.


```
{
  "SchemaVersion": "2018-10-08",
  "Id": "arn:aws:patchmanager:us-east-2:111122223333:instance/i-02573cafcfEXAMPLE/
document/AWS-RunPatchBaseline/run-command/d710f5bd-04e3-47b4-82f6-df4e0EXAMPLE",
  "ProductArn": "arn:aws:securityhub:us-east-1::product/aws/ssm-patch-manager",
  "GeneratorId": "d710f5bd-04e3-47b4-82f6-df4e0EXAMPLE",
  "AwsAccountId": "111122223333",
  "Types": [
    "Software & Configuration Checks/Patch Management/Compliance"
  ],
  "CreatedAt": "2021-11-11T22:05:25Z",
  "UpdatedAt": "2021-11-11T22:05:25Z",
  "Severity": {
    "Label": "INFORMATIONAL",
    "Normalized": 0
  },
  "Title": "Systems Manager Patch Summary - Managed Instance Non-Compliant",
  "Description": "This AWS control checks whether each instance that is managed by AWS
 Systems Manager is in compliance with the rules of the patch baseline that applies to
 that instance when a compliance Scan runs.",
  "Remediation": {
    "Recommendation": {
      "Text": "For information about bringing instances into patch compliance, see
 'Remediating out-of-compliance instances (Patch Manager)'.",
```

```
      "Url": "https://docs.aws.amazon.com/systems-manager/latest/userguide/patch-
compliance-remediation.html"
    }
  },
  "SourceUrl": "https://us-east-2.console.aws.amazon.com/systems-manager/managed-
instances/i-02573cafcfEXAMPLE/patch?region=us-east-2",
  "ProductFields": {
    "aws/securityhub/FindingId": "arn:aws:securityhub:us-east-2::product/aws/ssm-
patch-manager/arn:aws:patchmanager:us-east-2:111122223333:instance/i-02573cafcfEXAMPLE/
document/AWS-RunPatchBaseline/run-command/d710f5bd-04e3-47b4-82f6-df4e0EXAMPLE",
    "aws/securityhub/ProductName": "Systems Manager Patch Manager",
    "aws/securityhub/CompanyName": "AWS"
  },
  "Resources": [
    {
      "Type": "AwsEc2Instance",
      "Id": "i-02573cafcfEXAMPLE",
      "Partition": "aws",
      "Region": "us-east-2"
    }
  ],
  "WorkflowState": "NEW",
  "Workflow": {
    "Status": "NEW"
  },
  "RecordState": "ACTIVE",
  "PatchSummary": {
    "Id": "pb-0c10e65780EXAMPLE",
    "InstalledCount": 45,
    "MissingCount": 2,
    "FailedCount": 0,
    "InstalledOtherCount": 396,
    "InstalledRejectedCount": 0,
    "InstalledPendingReboot": 0,
    "OperationStartTime": "2021-11-11T22:05:06Z",
    "OperationEndTime": "2021-11-11T22:05:25Z",
    "RebootOption": "NoReboot",
    "Operation": "SCAN"
  }
}
```

Activation et configuration de l'intégration

Pour utiliser l'intégration de Patch Manager à Security Hub, vous devez activer Security Hub. Pour obtenir des informations sur l'activation de Security Hub, veuillez consulter Configuration de Security Hub dans le Guide de l'utilisateur AWS Security Hub .

La procédure suivante décrit l'intégration de Patch Manager et Security Hub lorsque Security Hub est déjà actif mais que l'intégration de Patch Manager est désactivée. Cette procédure doit être effectuée uniquement si l'intégration a été désactivée manuellement.

Pour ajouter l'intégration de Patch Manager et Security Hub

1. Dans le panneau de navigation, sélectionnez Patch Manager.
2. Sélectionnez l'onglet Settings.

   -ou-

   Si vous accédez à Patch Manager pour la première fois dans la Région AWS actuelle, choisissez Commencer par une présentation, puis sélectionnez l'onglet Paramètres.
3. Dans la section Exporter vers Security Hub, à droite de la case Les résultats de conformité des correctifs ne sont pas exportés vers Security Hub, sélectionnez Activer.

Comment arrêter l'envoi des résultats

Pour arrêter l'envoi des résultats à Security Hub, vous pouvez utiliser la console Security Hub ou l'API.

Pour plus d'informations, consultez les rubriques suivantes dans le AWS Security Hub Guide de l'utilisateur :

- Désactivation et activation du flux de résultats d'une intégration (console)
- Désactivation du flux de résultats d'une intégration (API Security Hub, AWS CLI)

Fonctionnement de Patch Manager (AWS CLI)

Cette section inclut des exemples de commandes d'AWS Command Line Interface (AWS CLI) que vous pouvez utiliser pour exécuter des tâches de configuration de Patch Manager, une des fonctionnalités de AWS Systems Manager.

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'un référentiel de correctifs personnalisée, consultez Didacticiel : application de correctifs à un environnement de serveur (AWS CLI).

Pour de plus amples informations sur l'utilisation de la AWS CLI pour des tâches AWS Systems Manager, veuillez consulter la section AWS Systems Manager de la AWS CLIRéférence des commandes.

Rubriques

- Commandes AWS CLI pour des référentiels de correctifs
- Commandes AWS CLI pour les groupes de correctifs
- Commandes AWS CLI pour afficher les résumés et les détails des correctifs
- Commandes AWS CLI pour analyser et corriger des nœuds gérés

Commandes AWS CLI pour des référentiels de correctifs

Exemples de commandes pour les référentiels de correctifs

- Créer un référentiel de correctifs
- Création d'un référentiel de correctifs avec des référentiels personnalisés pour les différentes versions du système d'exploitation
- Mettre à jour un référentiel de correctifs
- Renommer un référentiel de correctifs
- Supprimer un référentiel de correctifs
- Afficher toutes les références de correctifs
- Répertorier toutes les référentiels de correctifs fournis par AWS
- Répertorier mes références de correctifs
- Afficher un référentiel de correctifs
- Obtenir le référentiel de correctifs par défaut
- Définir un référentiel de correctifs personnalisée comme valeur par défaut
- Réinitialiser un référentiel de correctifs AWS comme valeur par défaut
- Baliser un référentiel de correctifs
- Répertorier les balises d'un référentiel de correctifs
- Supprimer une balise d'un référentiel de correctifs

Créer un référentiel de correctifs

La commande suivante crée un référentiel de correctifs approuvant 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, le référentiel de correctifs a été balisée pour indiquer qu'elle est destinée à un environnement de production.

Linux & macOS

```
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 Server

```
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 retourne des informations telles que les suivantes.

```
{
```

```
    "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

Création d'un référentiel de correctifs avec des référentiels personnalisés pour les différentes versions du système d'exploitation

S'applique uniquement aux nœuds gérés 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 autorisé par défaut sur Amazon Linux 2017.09, mais peut être adapté à un autre référentiel source configuré pour un nœud géré.

> **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 retourne des informations telles que les suivantes.

```
{
    "BaselineId": "pb-0c10e65780EXAMPLE"
}
```

Mettre à jour un référentiel de correctifs

La commande suivante ajoute deux correctifs comme refusés et un correctif comme approuvé à un référentiel de correctifs existante.

> **Note**
>
> Pour obtenir des informations sur les formats acceptés pour les listes de correctifs approuvés et de correctifs rejetés, veuillez consulter À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés.

Linux & macOS

```
aws ssm update-patch-baseline \
    --baseline-id pb-0c10e65780EXAMPLE \
    --rejected-patches "KB2032276" "MS10-048" \
    --approved-patches "KB2124261"
```

Windows Server

```
aws ssm update-patch-baseline ^
    --baseline-id pb-0c10e65780EXAMPLE ^
    --rejected-patches "KB2032276" "MS10-048" ^
    --approved-patches "KB2124261"
```

Le système retourne des informations telles que les suivantes.

```
{
   "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 un référentiel de correctifs

Linux & macOS

```
aws ssm update-patch-baseline \
```

```
    --baseline-id pb-0c10e65780EXAMPLE \
    --name "Windows-Server-2012-R2-Important-and-Critical-Security-Updates"
```

Windows Server

```
aws ssm update-patch-baseline ^
    --baseline-id pb-0c10e65780EXAMPLE ^
    --name "Windows-Server-2012-R2-Important-and-Critical-Security-Updates"
```

Le système retourne des informations telles que les suivantes.


```
{
   "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 un référentiel de correctifs

```
aws ssm delete-patch-baseline --baseline-id "pb-0c10e65780EXAMPLE"
```

Le système retourne des informations telles que les suivantes.

```
{
   "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

Afficher toutes les références de correctifs

```
aws ssm describe-patch-baselines
```

Le système retourne des informations telles que les suivantes.

```
{
   "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 référentiels de correctifs dans une Région AWS.

Linux & macOS

```
aws ssm describe-patch-baselines \
    --region us-east-2 \
    --filters "Key=OWNER,Values=[All]"
```

Windows Server

```
aws ssm describe-patch-baselines ^
    --region us-east-2 ^
    --filters "Key=OWNER,Values=[All]"
```

Le système retourne des informations telles que les suivantes.

```
{
   "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 référentiels de correctifs fournis par AWS

Linux & macOS

```
aws ssm describe-patch-baselines \
    --region us-east-2 \
    --filters "Key=OWNER,Values=[AWS]"
```

Windows Server

```
aws ssm describe-patch-baselines ^
    --region us-east-2 ^
    --filters "Key=OWNER,Values=[AWS]"
```

Le système retourne des informations telles que les suivantes.

```
{
   "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 & macOS

```
aws ssm describe-patch-baselines \
    --region us-east-2 \
```

```
    --filters "Key=OWNER,Values=[Self]"
```

Windows Server

```
aws ssm describe-patch-baselines ^
    --region us-east-2 ^
    --filters "Key=OWNER,Values=[Self]"
```

Le système retourne des informations telles que les suivantes.

```
{
   "BaselineIdentities":[
      {
         "BaselineName":"Windows-Server-2012R2",
         "DefaultBaseline":false,
         "BaselineDescription":"Windows Server 2012 R2, Important and Critical security
 updates",
         "BaselineId":"pb-0c10e65780EXAMPLE"
      }
   ]
}
```

Afficher un référentiel de correctifs

```
aws ssm get-patch-baseline --baseline-id pb-0c10e65780EXAMPLE
```

> **Note**
>
> Pour les référentiels de correctifs personnalisés, vous pouvez spécifier l'ID de référentiel de correctifs ou l'Amazon Resource Name (ARN) complet. Pour le référentiel de correctifs fournie par AWS, vous devez spécifier l'ARN complet. Par exemple, `arn:aws:ssm:us-east-2:075727635805:patchbaseline/pb-0c10e65780EXAMPLE`.

Le système retourne des informations telles que les suivantes.

```
{
   "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 le référentiel de correctifs par défaut

```
aws ssm get-default-patch-baseline --region us-east-2
```

Le système retourne des informations telles que les suivantes.

```
{
   "BaselineId":"arn:aws:ssm:us-east-2:111122223333:patchbaseline/pb-0c10e65780EXAMPLE"
}
```

Définir un référentiel de correctifs personnalisée comme valeur par défaut

Linux & macOS

```
aws ssm register-default-patch-baseline \
    --region us-east-2 \
    --baseline-id "pb-0c10e65780EXAMPLE"
```

Windows Server

```
aws ssm register-default-patch-baseline ^
    --region us-east-2 ^
    --baseline-id "pb-0c10e65780EXAMPLE"
```

Le système retourne des informations telles que les suivantes.

```
{
   "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

Réinitialiser un référentiel de correctifs AWS comme valeur par défaut

Linux & macOS

```
aws ssm register-default-patch-baseline \
    --region us-east-2 \
```

```
    --baseline-id "arn:aws:ssm:us-east-2:123456789012:patchbaseline/
pb-0c10e65780EXAMPLE"
```

Windows Server

```
aws ssm register-default-patch-baseline ^
    --region us-east-2 ^
    --baseline-id "arn:aws:ssm:us-east-2:123456789012:patchbaseline/
pb-0c10e65780EXAMPLE"
```

Le système retourne des informations telles que les suivantes.

```
{
   "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

Baliser un référentiel de correctifs

Linux & macOS

```
aws ssm add-tags-to-resource \
    --resource-type "PatchBaseline" \
    --resource-id "pb-0c10e65780EXAMPLE" \
    --tags "Key=Project,Value=Testing"
```

Windows Server

```
aws ssm add-tags-to-resource ^
    --resource-type "PatchBaseline" ^
    --resource-id "pb-0c10e65780EXAMPLE" ^
    --tags "Key=Project,Value=Testing"
```

Répertorier les balises d'un référentiel de correctifs

Linux & macOS

```
aws ssm list-tags-for-resource \
    --resource-type "PatchBaseline" \
    --resource-id "pb-0c10e65780EXAMPLE"
```

Windows Server

```
aws ssm list-tags-for-resource ^
    --resource-type "PatchBaseline" ^
    --resource-id "pb-0c10e65780EXAMPLE"
```

Supprimer une balise d'un référentiel de correctifs

Linux & macOS

```
aws ssm remove-tags-from-resource \
    --resource-type "PatchBaseline" \
    --resource-id "pb-0c10e65780EXAMPLE" \
    --tag-keys "Project"
```

Windows Server

```
aws ssm remove-tags-from-resource ^
    --resource-type "PatchBaseline" ^
    --resource-id "pb-0c10e65780EXAMPLE" ^
    --tag-keys "Project"
```

Commandes AWS CLI pour les groupes de correctifs

Exemples de commandes pour les groupes de correctifs

- Créer un groupe de correctifs
- Enregistrer un groupe de correctifs « Serveurs web » avec un référentiel de correctifs
- Enregistrer un groupe de correctifs « Backend » dans le référentiel de correctifs fourni par AWS
- Afficher les enregistrements du groupe de correctifs
- Annuler l'enregistrement d'un groupe de correctifs à partir d'un référentiel de correctifs

Créer un groupe de correctifs

Pour faciliter l'organisation des opérations d'application de correctifs, nous vous recommandons d'ajouter des nœuds gérés à des groupes de correctifs à l'aide de balises. Les groupes de correctifs nécessitent l'utilisation de la clé de balise `Patch Group` ou `PatchGroup`. Si vous avez autorisé les balises dans les métadonnées des instances EC2, vous devez utiliser `PatchGroup` (sans espace).

Toutefois, peu importe le choix de la valeur d'une balise, la clé de balise doit être `Patch Group` ou `PatchGroup`. Pour de plus amples informations sur les groupes de correctifs, consultez À propos des groupes de correctifs.

Après avoir regroupé vos nœuds gérés à l'aide de balises, vous devez ajouter la valeur du groupe de correctifs à un référentiel de correctifs. En enregistrant le groupe de correctifs auprès d'un référentiel 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 Elastic Compute Cloud (Amazon EC2) et AWS CLI, vous pouvez appliquer `Key = Patch Group` ou `Key = PatchGroup` des balises aux instances non configurées utilisables avec Systems Manager. Si une instance EC2 que vous prévoyez de voir dans Patch Manager n'est pas répertoriée après l'application de `Patch Group` ou `Key = PatchGroup` balise, consultez Résolution des problèmes de disponibilité des nœuds gérés afin d'obtenir des conseils de dépannage.

Exécutez la commande suivante pour ajouter la balise `PatchGroup` à une instance EC2.

```
aws ec2 create-tags --resources "i-1234567890abcdef0" --tags
 "Key=PatchGroup,Value=GroupValue"
```

Tâche 2 : Ajout de nœuds gérés à un groupe de correctifs à l'aide de balises

Exécutez la commande suivante pour ajouter la balise `PatchGroup` à un nœud géré.

Linux & macOS

```
aws ssm add-tags-to-resource \
    --resource-type "ManagedInstance" \
    --resource-id "mi-0123456789abcdefg" \
    --tags "Key=PatchGroup,Value=GroupValue"
```

Windows Server

```
aws ssm add-tags-to-resource ^
```

```
    --resource-type "ManagedInstance" ^
    --resource-id "mi-0123456789abcdefg" ^
    --tags "Key=PatchGroup,Value=GroupValue"
```

Tâche 3 : Ajout d'un groupe de correctifs à un référentiel de correctifs

Exécutez la commande suivante pour associer une valeur de balise `PatchGroup` au référentiel de correctifs spécifiée.

Linux & macOS

```
aws ssm register-patch-baseline-for-patch-group \
    --baseline-id "pb-0c10e65780EXAMPLE" \
    --patch-group "Development"
```

Windows Server

```
aws ssm register-patch-baseline-for-patch-group ^
    --baseline-id "pb-0c10e65780EXAMPLE" ^
    --patch-group "Development"
```

Le système retourne des informations telles que les suivantes.

```
{
  "PatchGroup": "Development",
  "BaselineId": "pb-0c10e65780EXAMPLE"
}
```

Enregistrer un groupe de correctifs « Serveurs web » avec un référentiel de correctifs

Linux & macOS

```
aws ssm register-patch-baseline-for-patch-group \
    --baseline-id "pb-0c10e65780EXAMPLE" \
    --patch-group "Web Servers"
```

Windows Server

```
aws ssm register-patch-baseline-for-patch-group ^
    --baseline-id "pb-0c10e65780EXAMPLE" ^
```

```
    --patch-group "Web Servers"
```

Le système retourne des informations telles que les suivantes.

```
{
   "PatchGroup":"Web Servers",
   "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

Enregistrer un groupe de correctifs « Backend » dans le référentiel de correctifs fourni par AWS

Linux & macOS

```
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 Server

```
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 retourne des informations telles que les suivantes.

```
{
   "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 retourne des informations telles que les suivantes.

```
{
   "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'un référentiel de correctifs

Linux & macOS

```
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 Server

```
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 retourne des informations telles que les suivantes.

```
{
   "PatchGroup":"Production",
   "BaselineId":"arn:aws:ssm:us-east-2:111122223333:patchbaseline/pb-0c10e65780EXAMPLE"
}
```

Commandes AWS CLI pour afficher les résumés et les détails des correctifs

Exemples de commandes pour afficher les résumés et les détails des correctifs

- Obtenir tous les correctifs définis par un référentiel de correctifs
- Obtenir tous les correctifs pour AmazOnLinux 2018.03 ayant une classification SECURITY et une sévérité de Critical
- Obtenir tous les correctifs pour Windows Server 2012 dont la sévérité MSRC est Critical
- Obtenir tous les correctifs disponibles
- Obtenir le résumé de l'état des correctifs par nœud géré
- Obtenir les informations de conformité des correctifs pour un nœud géré
- Afficher les résultats de conformité de l'application de correctifs (AWS CLI)

Obtenir tous les correctifs définis par un référentiel de correctifs

> **Note**
>
> Cette commande est seulement prise en charge pour les référentiels de correctifs de Windows Server.

Linux & macOS

```
aws ssm describe-effective-patches-for-patch-baseline \
    --region us-east-2 \
    --baseline-id "pb-0c10e65780EXAMPLE"
```

Windows Server

```
aws ssm describe-effective-patches-for-patch-baseline ^
    --region us-east-2 ^
```

```
    --baseline-id "pb-0c10e65780EXAMPLE"
```

Le système retourne des informations telles que les suivantes.

```
{
   "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---
```

Obtenir tous les correctifs pour AmazOnLinux 2018.03 ayant une classification **`SECURITY`** et une sévérité de **`Critical`**

Linux & macOS

```
aws ssm describe-available-patches \
    --region us-east-2 \
    --filters Key=PRODUCT,Values=AmazonLinux2018.03 Key=SEVERITY,Values=Critical
```

Windows Server

```
aws ssm describe-available-patches ^
    --region us-east-2 ^
    --filters Key=PRODUCT,Values=AmazonLinux2018.03 Key=SEVERITY,Values=Critical
```

Le système retourne des informations telles que les suivantes.

```
{
    "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 sévérité MSRC est **`Critical`**

Linux & macOS

```
aws ssm describe-available-patches \
    --region us-east-2 \
    --filters Key=PRODUCT,Values=WindowsServer2012 Key=MSRC_SEVERITY,Values=Critical
```

Windows Server

```
aws ssm describe-available-patches ^
    --region us-east-2 ^
    --filters Key=PRODUCT,Values=WindowsServer2012 Key=MSRC_SEVERITY,Values=Critical
```

Le système retourne des informations telles que les suivantes.

```
{
   "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 retourne des informations telles que les suivantes.

```
{
```

```
    "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---
```

Obtenir le résumé de l'état des correctifs par nœud géré

Le résumé par nœud géré présente un certain nombre de correctifs en indiquant les états suivants par nœud : « NotApplicable », « Missing », « Failed », « InstalledOther » et « Installed ».

Linux & macOS

```
aws ssm describe-instance-patch-states \
    --instance-ids i-08ee91c0b17045407 i-09a618aec652973a9
```

Windows Server

```
aws ssm describe-instance-patch-states ^
    --instance-ids i-08ee91c0b17045407 i-09a618aec652973a9
```

Le système retourne des informations telles que les suivantes.


```
{
   "InstancePatchStates":[
      {
            "InstanceId": "i-08ee91c0b17045407",
            "PatchGroup": "",
            "BaselineId": "pb-0c10e65780EXAMPLE",
            "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-0c10e65780EXAMPLE",
            "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 un nœud géré

```
aws ssm describe-instance-patches --instance-id i-08ee91c0b17045407
```

Le système retourne des informations telles que les suivantes.

```
{
   "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---
```

Afficher les résultats de conformité de l'application de correctifs (AWS CLI)

Pour afficher les résultats de conformité des correctifs pour un nœud géré individuel

Exécutez la commande suivante dans l'AWS Command Line Interface (AWS CLI) afin d'afficher les résultats de conformité des correctifs pour un nœud géré individuel.

```
aws ssm describe-instance-patch-states --instance-id instance-id
```

Remplacez *instance-id* par l'ID du nœud géré dont vous souhaitez afficher les résultats, au format `i-02573cafcfEXAMPLE` ou `mi-0282f7c436EXAMPLE`.

Les systèmes renvoient des informations telles que les suivantes.

```
{
    "InstancePatchStates": [
        {
            "InstanceId": "i-02573cafcfEXAMPLE",
            "PatchGroup": "mypatchgroup",
            "BaselineId": "pb-0c10e65780EXAMPLE",
            "SnapshotId": "a3f5ff34-9bc4-4d2c-a665-4d1c1EXAMPLE",
            "CriticalNonCompliantCount": 2,
            "SecurityNonCompliantCount": 2,
            "OtherNonCompliantCount": 1,
            "InstalledCount": 123,
            "InstalledOtherCount": 334,
            "InstalledPendingRebootCount": 0,
            "InstalledRejectedCount": 0,
            "MissingCount": 1,
            "FailedCount": 2,
            "UnreportedNotApplicableCount": 11,
            "NotApplicableCount": 2063,
            "OperationStartTime": "2021-05-03T11:00:56-07:00",
            "OperationEndTime": "2021-05-03T11:01:09-07:00",
            "Operation": "Scan",
```

```
            "LastNoRebootInstallOperationTime": "2020-06-14T12:17:41-07:00",
            "RebootOption": "RebootIfNeeded"
        }
    ]
}
```

Pour afficher un résumé du nombre de correctifs pour toutes les instances EC2 dans une région

La commande `describe-instance-patch-states` prend en charge la récupération des résultats pour une seule instance gérée à la fois. Cependant, l'utilisation d'un script personnalisé avec la commande `describe-instance-patch-states` vous permet de générer un rapport plus granulaire.

Par exemple, si l'outil de filtrage jq est installé sur votre machine locale, vous pouvez exécuter la commande suivante pour identifier celle de vos instances EC2 qui a un statut de `InstalledPendingReboot` dans une Région AWS particulière.

```
aws ssm describe-instance-patch-states \
    --instance-ids $(aws ec2 describe-instances --region region | jq
 '.Reservations[].Instances[] | .InstanceId' | tr '\n|"' ' ') \
    --output text --query 'InstancePatchStates[*].{Instance:InstanceId,
 InstalledPendingRebootCount:InstalledPendingRebootCount}'
```

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

Par exemple :

```
aws ssm describe-instance-patch-states \
    --instance-ids $(aws ec2 describe-instances --region us-east-2 | jq
 '.Reservations[].Instances[] | .InstanceId' | tr '\n|"' ' ') \
    --output text --query 'InstancePatchStates[*].{Instance:InstanceId,
 InstalledPendingRebootCount:InstalledPendingRebootCount}'
```

Le système retourne des informations telles que les suivantes.

```
1       i-02573cafcfEXAMPLE
0       i-0471e04240EXAMPLE
3       i-07782c72faEXAMPLE
```

```
6       i-083b678d37EXAMPLE
0       i-03a530a2d4EXAMPLE
1       i-01f68df0d0EXAMPLE
0       i-0a39c0f214EXAMPLE
7       i-0903a5101eEXAMPLE
7       i-03823c2fedEXAMPLE
```

Outre `InstalledPendingRebootCount`, la liste des types de nombres interrogeables comprend les éléments suivants :

- `CriticalNonCompliantCount`
- `SecurityNonCompliantCount`
- `OtherNonCompliantCount`
- `UnreportedNotApplicableCount`
- `InstalledPendingRebootCount`
- `FailedCount`
- `NotApplicableCount`
- `InstalledRejectedCount`
- `InstalledOtherCount`
- `MissingCount`
- `InstalledCount`

Commandes AWS CLI pour analyser et corriger des nœuds gérés

Après avoir exécuté les commandes suivantes pour analyser la conformité des correctifs ou installer des correctifs, vous pouvez utiliser les commandes de la section Commandes AWS CLI pour afficher les résumés et les détails des correctifs pour afficher des informations sur le statut et la conformité des correctifs.

Exemples de commandes

- Analyser les nœuds gérés pour vérifier la conformité des correctifs (AWS CLI)
- Installer des correctifs sur les nœuds gérés (AWS CLI)

Analyser les nœuds gérés pour vérifier la conformité des correctifs (AWS CLI)

Pour analyser les nœuds gérés afin de vérifier la conformité des correctifs

Exécutez la commande suivante.

Linux & macOS

```
aws ssm send-command \
    --document-name 'AWS-RunPatchBaseline' \
    --targets Key=InstanceIds,Values='i-02573cafcfEXAMPLE,i-0471e04240EXAMPLE' \
    --parameters 'Operation=Scan' \
    --timeout-seconds 600
```

Windows Server

```
aws ssm send-command ^
    --document-name "AWS-RunPatchBaseline" ^
    --targets Key=InstanceIds,Values="i-02573cafcfEXAMPLE,i-0471e04240EXAMPLE" ^
    --parameters "Operation=Scan" ^
    --timeout-seconds 600
```

Le système retourne des informations telles que les suivantes.

```
{
    "Command": {
        "CommandId": "a04ed06c-8545-40f4-87c2-a0babEXAMPLE",
        "DocumentName": "AWS-RunPatchBaseline",
        "DocumentVersion": "$DEFAULT",
        "Comment": "",
        "ExpiresAfter": 1621974475.267,
        "Parameters": {
            "Operation": [
                "Scan"
            ]
        },
        "InstanceIds": [],
        "Targets": [
            {
                "Key": "InstanceIds",
                "Values": [
                    "i-02573cafcfEXAMPLE,
                     i-0471e04240EXAMPLE"
                ]
            }
        ],
```

```
        "RequestedDateTime": 1621952275.267,
        "Status": "Pending",
        "StatusDetails": "Pending",
        "TimeoutSeconds": 600,

    ---output truncated---

    }
}
```

Pour analyser les nœuds gérés et vérifier la conformité des correctifs par balise de groupe de correctifs

Exécutez la commande suivante.

Linux & macOS

```
aws ssm send-command \
    --document-name 'AWS-RunPatchBaseline' \
    --targets Key='tag:PatchGroup',Values='Web servers' \
    --parameters 'Operation=Scan' \
    --timeout-seconds 600
```

Windows Server

```
aws ssm send-command ^
    --document-name "AWS-RunPatchBaseline" ^
    --targets Key="tag:PatchGroup",Values="Web servers" ^
    --parameters "Operation=Scan" ^
    --timeout-seconds 600
```

Le système retourne des informations telles que les suivantes.

```
{
    "Command": {
        "CommandId": "87a448ee-8adc-44e0-b4d1-6b429EXAMPLE",
        "DocumentName": "AWS-RunPatchBaseline",
        "DocumentVersion": "$DEFAULT",
        "Comment": "",
        "ExpiresAfter": 1621974983.128,
        "Parameters": {
```

```
            "Operation": [
                "Scan"
            ]
        },
        "InstanceIds": [],
        "Targets": [
            {
                "Key": "tag:PatchGroup",
                "Values": [
                    "Web servers"
                ]
            }
        ],
        "RequestedDateTime": 1621952783.128,
        "Status": "Pending",
        "StatusDetails": "Pending",
        "TimeoutSeconds": 600,

    ---output truncated---

    }
}
```

Installer des correctifs sur les nœuds gérés (AWS CLI)

Pour installer des correctifs sur des nœuds gérés spécifiques

Exécutez la commande suivante.

> **Note**
>
> Si nécessaire, les nœuds gérés cibles redémarrent pour finaliser l'installation des correctifs. Pour de plus amples informations, veuillez consulter À propos du document SSM `AWS-RunPatchBaseline`.

Linux & macOS

```
aws ssm send-command \
    --document-name 'AWS-RunPatchBaseline' \
    --targets Key=InstanceIds,Values='i-02573cafcfEXAMPLE,i-0471e04240EXAMPLE' \
    --parameters 'Operation=Install' \
```

```
    --timeout-seconds 600
```

Windows Server

```
aws ssm send-command ^
    --document-name "AWS-RunPatchBaseline" ^
    --targets Key=InstanceIds,Values="i-02573cafcfEXAMPLE,i-0471e04240EXAMPLE" ^
    --parameters "Operation=Install" ^
    --timeout-seconds 600
```

Le système retourne des informations telles que les suivantes.

```
{
    "Command": {
        "CommandId": "5f403234-38c4-439f-a570-93623EXAMPLE",
        "DocumentName": "AWS-RunPatchBaseline",
        "DocumentVersion": "$DEFAULT",
        "Comment": "",
        "ExpiresAfter": 1621975301.791,
        "Parameters": {
            "Operation": [
                "Install"
            ]
        },
        "InstanceIds": [],
        "Targets": [
            {
                "Key": "InstanceIds",
                "Values": [
                    "i-02573cafcfEXAMPLE,
                     i-0471e04240EXAMPLE"
                ]
            }
        ],
        "RequestedDateTime": 1621953101.791,
        "Status": "Pending",
        "StatusDetails": "Pending",
        "TimeoutSeconds": 600,

    ---output truncated---

    }
```

```
}
```

Pour installer des correctifs sur des nœuds gérés appartenant à un groupe de correctifs spécifique

Exécutez la commande suivante.

Linux & macOS

```
aws ssm send-command \
    --document-name 'AWS-RunPatchBaseline' \
    --targets Key='tag:PatchGroup',Values='Web servers' \
    -parameters 'Operation=Install' \
    --timeout-seconds 600
```

Windows Server

```
aws ssm send-command ^
    --document-name "AWS-RunPatchBaseline" ^
    --targets Key="tag:PatchGroup",Values="Web servers" ^
    --parameters "Operation=Install" ^
    --timeout-seconds 600
```

Le système retourne des informations telles que les suivantes.

```
{
    "Command": {
        "CommandId": "fa44b086-7d36-4ad5-ac8d-627ecEXAMPLE",
        "DocumentName": "AWS-RunPatchBaseline",
        "DocumentVersion": "$DEFAULT",
        "Comment": "",
        "ExpiresAfter": 1621975407.865,
        "Parameters": {
            "Operation": [
                "Install"
            ]
        },
        "InstanceIds": [],
        "Targets": [
            {
                "Key": "tag:PatchGroup",
                "Values": [
```

```
                    "Web servers"
                ]
            }
        ],
        "RequestedDateTime": 1621953207.865,
        "Status": "Pending",
        "StatusDetails": "Pending",
        "TimeoutSeconds": 600,

    ---output truncated---

    }
}
```

AWS Systems ManagerPatch Managertutoriels

Les didacticiels de cette section expliquent comment utiliser Patch Manager, une des fonctionnalités de AWS Systems Manager, pour une sélection de scénarios d'application de correctifs.

Rubriques

- Didacticiel : créer un référentiel de correctifs pour l'installation des Service Packs Windows (console)
- Didacticiel : mettre à jour les dépendances des applications, appliquer des correctifs sur un nœud géré et effectuer une surveillance de l'état spécifique à l'application
- Didacticiel : application de correctifs à un environnement de serveur (AWS CLI)

Didacticiel : créer un référentiel de correctifs pour l'installation des Service Packs Windows (console)

Lorsque vous créez un référentiel 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 grâce Patch Manager à une fonctionnalité de AWS Systems Manager, à condition que la mise à jour soit disponible dans Windows Update ou Windows Server Update Services (WSUS).

Vous pouvez configurer un référentiel 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 ci-dessous afin de créer un référentiel de correctifs personnalisé à utiliser exclusivement pour installer tous les Service Packs sur vos nœuds gérés Windows.

Créer un référentiel de correctifs pour l'installation des Service Packs Windows (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Patch Manager.
3. Choisissez l'onglet Référentiels de correctifs, puis choisissez Créer un référentiel de correctifs.
4. Pour Nom, entrez un nom pour votre nouvelle référentiel de correctifs, par exemple : `MyWindowsServicePackPatchBaseline`.
5. (Facultatif) Pour Description, saisissez une description pour cette référence de correctif.
6. Pour Système d'exploitation, sélectionnez `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.

   > **Note**
   >
   > Cette option n'est disponible que si vous avez accédé à Patch Manager pour la première fois avant la publication des politiques de correctifs le 22 décembre 2022.
   > Pour de plus amples informations, sur la définition d'un référentiel de correctifs existante en tant que référence par défaut, veuillez consulter Définition d'un référentiel de correctifs existante en tant que valeur par défaut.

8. Dans la section Règles d'approbation pour les systèmes d'exploitation), utilisez les champs pour créer une ou plusieurs règles d'approbation automatique.

   - Produits : versions des systèmes d'exploitation auxquelles s'applique la règle d'approbation, par exemple `WindowsServer2012`. Vous pouvez choisir une, plusieurs ou toutes les versions prises en charge de Windows. La sélection par défaut est `All`.
   - Classification : sélectionnez `ServicePacks`.

- Importance : valeur d'importance des correctifs à laquelle la règle va s'appliquer. Pour vous assurer que tous les Service Packs sont inclus dans la règle, sélectionnez `All`.
- Approbation automatique : méthode de sélection des patchs pour approbation automatique.
  - Approuver les correctifs après un nombre de jours spécifié : pendant lesquels Patch Manager doit attendre après la publication ou la mise à jour d'un correctif avant son approbation automatique. Vous pouvez entrer tout nombre entier situé entre zéro (0) et 360. Nous vous recommandons, en règle générale, de ne pas attendre plus de 100 jours.
  - L'approbation des correctifs publiés jusqu'à une date spécifique: date de publication des correctifs pour laquelle Patch Manager applique automatiquement tous les correctifs publiés à cette date ou avant cette date. Par exemple, si vous spécifiez le 7 juillet 2023, aucun correctif publié ou mis à jour le 8 juillet 2023 ou après ne sera installé automatiquement.
- (Facultatif) Rapport de conformité : niveau d'importance que vous voulez affecter aux Service Packs approuvés par la référence, par exemple `High`.

> **Note**
>
> Si vous spécifiez un niveau de rapport de conformité et que l'état des correctifs d'un Service Pack approuvé est indiqué `Missing`, le niveau de sévérité de conformité global indiqué pour le référentiel de correctifs est le niveau de sévérité que vous avez spécifié.

9. (Facultatif) Pour Gérer les balises, appliquez une ou plusieurs paires nom/valeur de clé de balise au référentiel 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 ce référentiel 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. Sélectionnez Créer un référentiel de correctif.

Didacticiel : mettre à jour les dépendances des applications, appliquer des correctifs sur un nœud géré et effectuer une surveillance de l'état spécifique à l'application

Dans bien des cas, un nœud géré doit être redémarré après l'installation de la dernière mise à jour logicielle. Toutefois, le redémarrage d'un nœud en production sans mesures de protection peut occasionner différents problèmes, comme le déclenchement d'alarmes, l'enregistrement de données métriques incorrectes et l'interruption des synchronisations de données.

Ce didacticiel illustre la façon d'éviter les problèmes de ce genre en utilisant le document AWS Systems Manager (document SSM) `AWS-RunPatchBaselineWithHooks` pour réaliser une opération complexe d'application de correctifs en plusieurs étapes qui effectue les opérations suivantes :

1. Empêcher de nouvelles connexions à l'application
2. Installer les mises à jour du système d'exploitation
3. Mettre à jour les dépendances de package de l'application
4. Redémarrer le système
5. Effectuer une surveillance de l'état spécifique à l'application

Pour cet exemple, nous avons configuré notre infrastructure de la façon suivante :

- Les machines virtuelles ciblées sont enregistrées en tant que nœuds gérés auprès de Systems Manager.
- `Iptables` est utilisé comme pare-feu local.
- L'application hébergée sur les nœuds gérés s'exécute sur le port 443.
- L'application hébergée sur les nœuds gérés est une application `nodeJS`.
- L'application hébergée sur les nœuds gérés est gérée par le gestionnaire de processus pm2.
- L'application dispose déjà d'un point de terminaison de surveillance de l'état spécifié.
- Le point de terminaison de surveillance de l'état de l'application n'exige aucune authentification de l'utilisateur final. Le point de terminaison autorise une surveillance de l'état qui répond aux exigences de l'organisation en matière d'établissement de la disponibilité. (Dans vos environnements, il pourrait suffire de vérifier simplement que l'application `nodeJS` est en cours d'exécution et peut écouter les demandes. Dans d'autres cas, vous pouvez également vouloir vérifier qu'une connexion à la couche de mise en cache ou à la couche de base de données est déjà établie.)

Les exemples de ce didacticiel sont fournis à titre indicatif uniquement. Ils ne sont pas destinés à être mis en œuvre en l'état dans les environnements de production. N'oubliez pas non plus que la fonction de hooks de cycle de vie de Patch Manager, une fonctionnalité de Systems Manager, avec le document `AWS-RunPatchBaselineWithHooks` peut prendre en charge de nombreux autres scénarios. Voici quelques exemples.

- Arrêtez l'agent de génération de rapport de métriques avant d'appliquer les correctifs, et relancez-le après le redémarrage du nœud géré.
- Détachez le nœud géré du cluster CRM ou PCS avant d'appliquer les correctifs, et attachez-le de nouveau après le redémarrage du nœud.
- Mettez à jour les logiciels tiers (comme les applications Adobe, Java, Tomcat, etc.) sur les machines Windows Server après l'application des mises à jour du système d'exploitation, mais avant le redémarrage du nœud géré.

Pour mettre à jour les dépendances des applications, appliquer des correctifs sur un nœud géré et effectuer une surveillance de l'état spécifique à l'application

1. Créez un document SSM avec le contenu suivant pour votre script de préinstallation et nommez-le `NodeJSAppPrePatch`. Remplacez *`your_application`* par le nom de votre application.

   Ce script bloque immédiatement les nouvelles demandes entrantes et fournit aux demandes déjà actives un délai de cinq secondes avant de commencer l'opération d'application de correctifs. Pour l'option `sleep`, spécifiez un nombre de secondes supérieur à ce qu'il faut habituellement aux requêtes entrantes pour s'effectuer.

   ```
   # exit on error
   set -e
   # set up rule to block incoming traffic
   iptables -I INPUT -j DROP -p tcp --syn --destination-port 443 || exit 1
   # wait for current connections to end. Set timeout appropriate to your
    application's latency
   sleep 5
   # Stop your application
   pm2 stop your_application
   ```

   Pour de plus amples informations sur la création de documents SSM, consultez Création du contenu du document SSM.

2. Créez un autre document SSM avec le contenu suivant pour votre script postinstallation, pour mettre à jour vos dépendances d'application, et nommez-le `NodeJSAppPostPatch`. Remplacez *`/your/application/path`* par le chemin d'accès à votre application.

```
cd /your/application/path
npm update
# you can use npm-check-updates if you want to upgrade major versions
```

3. Créez un autre document SSM avec le contenu suivant pour votre script `onExit`, pour rétablir votre application et effectuer une surveillance de l'état. Nommez ce document SSM `NodeJSApp0nExitPatch`. Remplacez *`your_application`* par le nom de votre application.

```
# exit on error
set -e
# restart nodeJs application
pm2 start your_application
# sleep while your application starts and to allow for a crash
sleep 10
# check with pm2 to see if your application is running
pm2 pid your_application
# re-enable incoming connections
iptables -D INPUT -j DROP -p tcp --syn --destination-port
# perform health check
/usr/bin/curl -m 10 -vk -A "" http://localhost:443/health-check || exit 1
```

4. Créez une associationState Manager, une capacité de AWS Systems Manager, pour lancer l'opération en effectuant les étapes suivantes :

   1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
   2. Dans le panneau de navigation, sélectionnez State Manager, puis Créer une association.
   3. Pour Nom, fournissez un nom permettant d'identifier le but de l'association.
   4. Dans la liste Document, sélectionnez `AWS-RunPatchBaselineWithHooks`.
   5. Pour Operation (Opération), sélectionnez Install (Installer).
   6. (Facultatif) Pour ID d'instantané, fournissez un GUID que vous générez pour accélérer l'opération et garantir la cohérence. La valeur du GUID peut être aussi simple que `00000000-0000-0000-0000-111122223333`.
   7. Pour Nom de document du hook de préinstallation, saisissez `NodeJSAppPrePatch`.

8. Pour Nom de document du hook de postinstallation, saisissez `NodeJSAppPostPatch`.
9. Pour On ExitHook Doc Name, entrez`NodeJSApp0nExitPatch`.

5. Dans Targets (Cibles), identifiez vos nœuds gérés en spécifiant des balises, en choisissant manuellement les nœuds, en choisissant un groupe de ressources ou en choisissant tous les nœuds gérés.
6. Pour Spécifier le calendrier, spécifiez la fréquence d'exécution de l'association. La fréquence d'application des correctifs sur les nœuds gérés est couramment définie sur une fois par semaine.
7. Dans la section Rate control (Contrôle du débit), sélectionnez les options permettant de contrôler la façon dont l'association s'exécute sur plusieurs nœuds gérés. Assurez-vous que seule une partie des nœuds gérés est mise à jour à la fois. Autrement, la totalité ou la majeure partie de votre flotte pourrait être déconnectée en même temps. 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.
8. (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) Saisissez les noms de compartiment et de préfixe (dossier) dans les zones.

   > **Note**
   >
   > Les autorisations S3 qui donnent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué au nœud géré, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, vérifiez que le profil d'instance ou le rôle de service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

9. Sélectionnez Create Association (Créer une association).

Didacticiel : application de correctifs à un environnement de serveur (AWS CLI)

La procédure suivante décrit l'application de correctifs à un environnement de serveur à l'aide d'un référentiel de correctifs personnalisée, de groupes de correctifs et d'une fenêtre de maintenance.

Avant de commencer

- Installez ou mettez à jour SSM Agent sur vos nœuds gérés. Pour appliquer des correctifs à des nœuds gérés Linux, ces derniers doivent exécuter SSM Agent version 2.0.834.0 ou ultérieure. Pour de plus amples informations, veuillez consulter Mise à jour de SSM Agent à l'aide de Run Command.
- Configurez les rôles et les autorisations pour la fonctionnalité Maintenance Windows, une des fonctionnalités de AWS Systems Manager. Pour de plus amples informations, veuillez consulter Configuration de Maintenance Windows.
- Si vous ne l'avez pas déjà fait, installez et configurez l'AWS Command Line Interface (AWS CLI).

  Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l'AWS CLI.

Pour configurer Patch Manager et appliquer des correctifs aux nœuds gérés (ligne de commande)

1. Exécutez la commande suivante pour créer un référentiel de correctifs pour Windows nommée `Production-Baseline`. Ce référentiel de correctifs approuve les correctifs pour un environnement de production 7 jours après leur publication ou leur dernière mise à jour. Cela signifie que le référentiel 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 nœuds gérés cibles auxquels s'applique le référentiel de correctifs. Pour de plus amples informations, consultez OperatingSystem et PatchFilter.

   Linux & macOS

   ```
   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,Impor
   ```

```
{Key=CLASSIFICATION,Values=[SecurityUpdates,Updates,ServicePacks,UpdateRollups,CriticalU
 \
    --description "Baseline containing all updates approved for production
 systems"
```

Windows Server

```
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,Impor
{Key=CLASSIFICATION,Values=[SecurityUpdates,Updates,ServicePacks,UpdateRollups,CriticalU
 ^
    --description "Baseline containing all updates approved for production
 systems"
```

Le système retourne des informations telles que les suivantes.

```
{
   "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

2. Exécutez les commandes suivantes pour enregistrer le référentiel de correctifs « Production-Baseline » pour deux groupes de correctifs. Les groupes sont nommés « serveurs de base de données » et « serveurs front-end ».

   Linux & macOS

   ```
   aws ssm register-patch-baseline-for-patch-group \
       --baseline-id pb-0c10e65780EXAMPLE \
       --patch-group "Database Servers"
   ```

   Windows Server

   ```
   aws ssm register-patch-baseline-for-patch-group ^
       --baseline-id pb-0c10e65780EXAMPLE ^
       --patch-group "Database Servers"
   ```

Le système retourne des informations telles que les suivantes.

```
{
   "PatchGroup":"Database Servers",
   "BaselineId":"pb-0c10e65780EXAMPLE"
}
```

Linux & macOS

```
aws ssm register-patch-baseline-for-patch-group \
    --baseline-id pb-0c10e65780EXAMPLE \
    --patch-group "Front-End Servers"
```

Windows Server

```
aws ssm register-patch-baseline-for-patch-group ^
    --baseline-id pb-0c10e65780EXAMPLE ^
    --patch-group "Front-End Servers"
```

Le système retourne des informations telles que les suivantes.

```
{
   "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 & macOS

   ```
   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 Server

```
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 retourne des informations telles que les suivantes.

```
{
   "WindowId":"mw-0c50858d01EXAMPLE"
}
```

Linux & macOS

```
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 Server

```
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 retourne des informations telles que les suivantes.

```
{
   "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 & macOS

   ```
   aws ssm register-target-with-maintenance-window \
       --window-id mw-0c50858d01EXAMPLE \
       --targets "Key=tag:PatchGroup,Values=Database Servers" \
       --owner-information "Database Servers" \
       --resource-type "INSTANCE"
   ```

   Windows Server

   ```
   aws ssm register-target-with-maintenance-window ^
       --window-id mw-0c50858d01EXAMPLE ^
       --targets "Key=tag:PatchGroup,Values=Database Servers" ^
       --owner-information "Database Servers" ^
       --resource-type "INSTANCE"
   ```

   Le système retourne des informations telles que les suivantes.

   ```
   {
      "WindowTargetId":"e32eecb2-646c-4f4b-8ed1-205fbEXAMPLE"
   }
   ```

   Linux & macOS

   ```
   aws ssm register-target-with-maintenance-window \
   --window-id mw-9a8b7c6d5eEXAMPLE \
   --targets "Key=tag:PatchGroup,Values=Front-End Servers" \
   --owner-information "Front-End Servers" \
   --resource-type "INSTANCE"
   ```

Windows Server

```
aws ssm register-target-with-maintenance-window ^
    --window-id mw-9a8b7c6d5eEXAMPLE ^
    --targets "Key=tag:PatchGroup,Values=Front-End Servers" ^
    --owner-information "Front-End Servers" ^
    --resource-type "INSTANCE"
```

Le système retourne des informations telles que les suivantes.

```
{
   "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 & macOS

```
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::123456789012:role/MW-Role" \
    --task-type "RUN_COMMAND" \
    --max-concurrency 2 \
    --max-errors 1 \
    --priority 1 \
    --task-invocation-parameters "RunCommand={Parameters={Operation=Install}}"
```

   Windows Server

```
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::123456789012:role/MW-Role" ^
```

```
    --task-type "RUN_COMMAND" ^
    --max-concurrency 2 ^
    --max-errors 1 ^
    --priority 1 ^
    --task-invocation-parameters "RunCommand={Parameters={Operation=Install}}"
```

Le système retourne des informations telles que les suivantes.

```
{
   "WindowTaskId":"4f7ca192-7e9a-40fe-9192-5cb15EXAMPLE"
}
```

Linux & macOS

```
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::123456789012:role/MW-Role" \
    --task-type "RUN_COMMAND" \
    --max-concurrency 2 \
    --max-errors 1 \
    --priority 1 \
    --task-invocation-parameters "RunCommand={Parameters={Operation=Install}}"
```

Windows Server

```
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::123456789012:role/MW-Role" ^
    --task-type "RUN_COMMAND" ^
    --max-concurrency 2 ^
    --max-errors 1 ^
    --priority 1 ^
    --task-invocation-parameters "RunCommand={Parameters={Operation=Install}}"
```

Le système retourne des informations telles que les suivantes.

```
{
   "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 récapitulatif détaillé de conformité des correctifs comprend le nombre de nœuds gérés et présente les correctifs en indiquant leurs états respectifs.

> **Note**
>
> Ce récapitulatif doit théoriquement afficher des zéros pour le nombre de nœuds gérés jusqu'à ce que la tâche d'application des correctifs soit exécutée lors de la première fenêtre de maintenance.

Linux & macOS

```
aws ssm describe-patch-group-state \
    --patch-group "Database Servers"
```

Windows Server

```
aws ssm describe-patch-group-state ^
    --patch-group "Database Servers"
```

Le système retourne des informations telles que les suivantes.

```
{
   "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 récapitulatif des états des correctifs par nœud géré pour un groupe de correctifs. Le récapitulatif par nœud géré présente un certain nombre de correctifs en indiquant leurs états respectifs par nœud géré pour un groupe de correctifs.

   Linux & macOS

   ```
   aws ssm describe-instance-patch-states-for-patch-group \
       --patch-group "Database Servers"
   ```

   Windows Server

   ```
   aws ssm describe-instance-patch-states-for-patch-group ^
       --patch-group "Database Servers"
   ```

   Le système retourne des informations telles que les suivantes.

   ```
   {
      "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 d'autres exemples de commandes AWS CLI que vous pouvez utiliser pour vos tâches de configuration du Patch Manager, consultez Fonctionnement de Patch Manager (AWS CLI).

Résolution des problèmes de Patch Manager

Utilisez les informations suivantes pour vous aider à résoudre les problèmes liés Patch Manager à une fonctionnalité de AWS Systems Manager.

Rubriques

- Problème : erreur « Invoke- PatchBaselineOperation : accès refusé » ou erreur « Impossible de télécharger le fichier depuis S3 » pour baseline_overrides.json
- Problème : le correctif échoue sans cause apparente ni message d'erreur
- Problème : résultats de conformité aux correctifs inattendus
- Erreurs lors de l'exécution de AWS-RunPatchBaseline sur Linux
- Erreurs lors de l'exécution de AWS-RunPatchBaseline sur Windows Server
- Contacter AWS Support

Problème : erreur « Invoke- PatchBaselineOperation : accès refusé » ou erreur « Impossible de télécharger le fichier depuis S3 » pour **baseline_overrides.json**

Problème : lorsque les opérations de correction spécifiées par votre politique de correction s'exécutent, vous recevez une erreur similaire à l'exemple suivant.

Example error on Windows Server

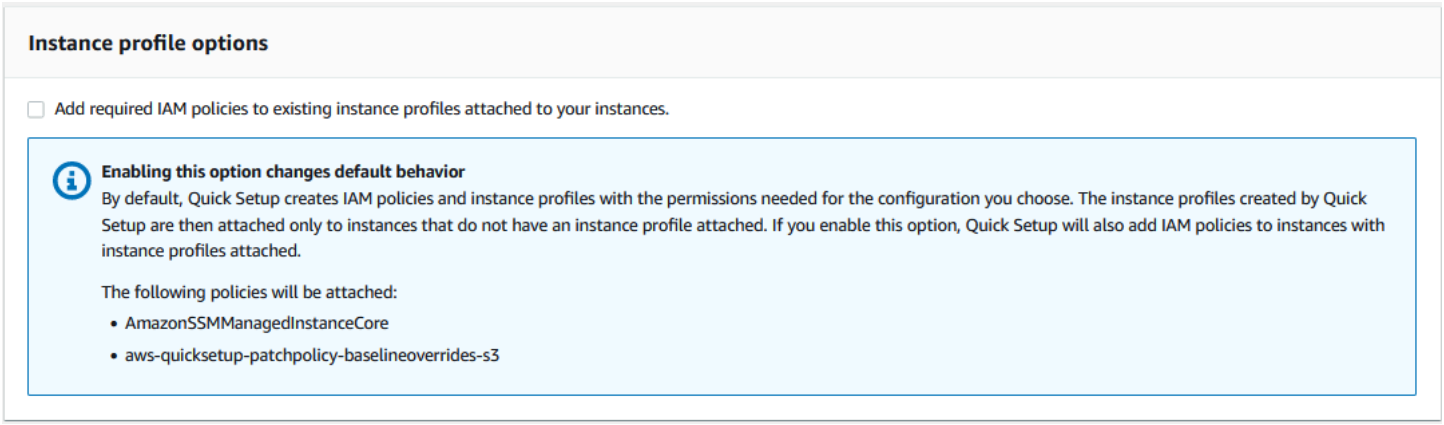
```
----------ERROR-------
Invoke-PatchBaselineOperation : Access Denied
At C:\ProgramData\Amazon\SSM\InstanceData\i-02573cafcfEXAMPLE\document\orchestr
ation\792dd5bd-2ad3-4f1e-931d-abEXAMPLE\PatchWindows\_script.ps1:219 char:13
+ $response = Invoke-PatchBaselineOperation -Operation Install -Snapsho ...
+ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```

```
+ CategoryInfo : OperationStopped: (Amazon.Patch.Ba...UpdateOpera
tion:InstallWindowsUpdateOperation) [Invoke-PatchBaselineOperation], Amazo
nS3Exception
+ FullyQualifiedErrorId : PatchBaselineOperations,Amazon.Patch.Baseline.Op
erations.PowerShellCmdlets.InvokePatchBaselineOperation
failed to run commands: exit status 0xffffffff
```

Example error on Linux

```
[INFO]: Downloading Baseline Override from s3://aws-quicksetup-
patchpolicy-123456789012-abcde/baseline_overrides.json
[ERROR]: Unable to download file from S3: s3://aws-quicksetup-
patchpolicy-123456789012-abcde/baseline_overrides.json.
[ERROR]: Error loading entrance module.
```

Cause : vous avez créé une politique de correctifs dans Quick Setup, et certains de vos nœuds gérés avaient déjà un profil d'instance attaché (pour les instances EC2) ou à une fonction du service (pour les machines non EC2). Cependant, vous n'avez pas coché la case Ajouter les politiques IAM requises aux profils d'instance existants attachés à vos instances, comme le montre l'image suivante.



Lorsque vous créez une politique de correctifs, un compartiment Amazon S3 est également créé pour stocker le fichier `baseline_overrides.json` de configuration de la politique. Si vous ne cochez pas la case Ajouter les politiques IAM requises aux profils d'instance existants attachés à vos instances lors de la création de la politique, les politiques IAM et les balises de ressources nécessaires pour accéder à `baseline_overrides.json` dans le compartiment S3 ne sont pas automatiquement ajoutées à vos profils d'instance et fonction du service IAM existants.

Solution 1 : supprimez la configuration de la politique de correctifs existante, puis créez-en une nouvelle, en veillant à cocher la case Ajouter les politiques IAM requises aux profils d'instance

existants attachés à vos instances. Cette sélection applique les politiques IAM créées par cette configuration Quick Setup aux nœuds auxquels un profil d'instance ou une fonction du service est déjà attaché. (Par défaut, Quick Setup ajoute les politiques requises aux instances et aux nœuds qui n'ont pas encore de profils d'instance ou de fonction du service.) Pour plus d'informations, consultez Automatiser l'application de correctifs à l'échelle de l'organisation à l'aide d'une politique de correctifs Quick Setup.

Solution 2 : ajoutez manuellement les autorisations et les balises requises à chaque profil d'instance IAM et à chaque fonction du service IAM que vous utilisez avec Quick Setup. Pour obtenir des instructions, veuillez consulter Autorisations pour le compartiment S3 de la politique de correctifs.

Problème : le correctif échoue sans cause apparente ni message d'erreur

Problème : une opération de correctif échoue sans renvoyer de message d'erreur.

Cause possible : si plusieurs invocations de `AWS-RunPatchBaseline` se produisent en même temps, elles peuvent entrer en conflit les unes avec les autres, ce qui entraîne l'échec des tâches de correctif. Cela peut ne pas être indiqué dans les journaux correctifs.

Pour vérifier si des opérations de correction simultanées peuvent s'être interrompues, consultez l'historique des commandes dansRun Command, une fonctionnalité de AWS Systems Manager. Pour un nœud géré qui présente un échec de correction, vérifiez si plusieurs opérations ont tenté de corriger l'ordinateur à moins de deux minutes d'intervalle. Ce scénario peut parfois provoquer une défaillance.

Vous pouvez également utiliser le AWS Command Line Interface (AWS CLI) pour vérifier les tentatives d'application de correctifs simultanées à l'aide de la commande suivante. Remplacez la valeur de *node-id* par l'ID de votre nœud géré.

```
aws ssm list-commands \
    --filter "key=DocumentName,value=AWS-RunPatchBaseline" \
    --query 'Commands[*].
{CommandId:CommandId,RequestedDateTime:RequestedDateTime,Status:Status}' \
    --instance-id node-id \
    --output table
```

Solution : si vous déterminez que la correction a échoué en raison d'opérations de correction concurrentes sur le même nœud géré, ajustez vos configurations de correction pour éviter que cela ne se reproduise. Par exemple, si deux fenêtres de maintenance spécifient des heures de correctif qui se chevauchent, supprimez ou révisez l'une d'entre elles. Si une fenêtre de maintenance spécifie

une opération de correction, mais qu'une politique de correctifs en spécifie une autre pour la même heure, envisagez de supprimer la tâche de la fenêtre de maintenance.

Si vous déterminez que les opérations de correctifs conflictuelles n'étaient pas la cause de l'échec dans ce scénario, nous vous recommandons de contacter AWS Support.

Problème : résultats de conformité aux correctifs inattendus

Problème : lorsque vous examinez les informations de conformité aux correctifs générées après une opération `Scan`, les résultats incluent des informations qui ne reflètent pas les règles définies dans votre référentiel de correctifs. Par exemple, une exception que vous avez ajoutée à la liste Rejected patches (Correctifs rejetés) dans un référentiel de correctifs est répertoriée comme `Missing`. Ou les correctifs considérés comme `Important` sont répertoriés comme manquants alors que votre référentiel de correctifs ne spécifie que des correctifs `Critical`.

Cause : Patch Manager prend actuellement en charge plusieurs méthodes d'exécution des opérations `Scan` :

- Une politique de correctifs configurée dans Quick Setup
- Une option de gestion des hôtes configurée dans Quick Setup
- Une fenêtre de maintenance pour exécuter un correctif `Scan` ou une tâche `Install`
- Une opération Patch now (Appliquer les correctifs maintenant) à la demande

Lorsqu'une opération `Scan` s'exécute, elle remplace les informations de conformité issues de l'analyse la plus récente. Si plusieurs méthodes sont configurées pour exécuter une opération `Scan` et qu'elles utilisent des référentiels de correctifs différents avec des règles différentes, elles se traduiront par des résultats de conformité aux correctifs différents.

Solution : pour éviter des résultats inattendus de conformité aux correctifs, nous vous recommandons de n'utiliser qu'une seule méthode à la fois pour exécuter l'opération `Scan` de Patch Manager. Pour plus d'informations, consultez Éviter les remplacements involontaires des données de conformité aux correctifs.

Erreurs lors de l'exécution de **`AWS-RunPatchBaseline`** sur Linux

Rubriques

Problème : erreur indiquant « Pas de fichier ou de répertoire »

Problème : lorsque vous exécutez `AWS-RunPatchBaseline`, l'application des correctifs échoue avec l'une des erreurs suivantes.

```
IOError: [Errno 2] No such file or directory: 'patch-baseline-operations-X.XX.tar.gz'
```

```
Unable to extract tar file: /var/log/amazon/ssm/patch-baseline-operations/patch-
baseline-operations-1.75.tar.gz.failed to run commands: exit status 155
```

```
Unable to load and extract the content of payload, abort.failed to run commands: exit
 status 152
```

Cause 1 : deux commandes servant à exécuter `AWS-RunPatchBaseline` s'exécutaient en même temps sur le même nœud géré. Cela crée une condition de concurrence qui empêche la création des `file patch-baseline-operations*` temporaires, ou l'accès normal à celles-ci.

Cause 2 : l'espace de stockage restant dans le répertoire `/var` est insuffisant.

Solution 1 : vérifiez que, pour une fenêtre de maintenance, deux tâches Run Command ou plus n'exécutent pas `AWS-RunPatchBaseline` avec le même niveau de priorité, ni sur les mêmes ID cibles. Si tel est le cas, réorganisez la priorité .Run Command est une fonctionnalité de AWS Systems Manager.

Solution 2 : vérifiez qu'une seule fenêtre de maintenance à la fois exécute des tâches Run Command qui utilisent `AWS-RunPatchBaseline` sur les mêmes cibles et selon le même calendrier. Si tel est le cas, modifiez le calendrier.

Solution 3 : vérifiez qu'une seule association State Manager exécute `AWS-RunPatchBaseline` selon le même calendrier et cible les mêmes nœuds gérés. State Manager est une fonctionnalité d' AWS Systems Manager.

Solution 4 : libérez suffisamment d'espace de stockage dans le répertoire `/var` pour les packages de mise à jour.

Problème : erreur indiquant « un autre processus a acquis yum lock »

Problème : lorsque vous exécutez `AWS-RunPatchBaseline`, l'application des correctifs échoue avec l'erreur suivante.

```
12/20/2019 21:41:48 root [INFO]: another process has acquired yum lock, waiting 2 s and
 retry.
```

Cause : le document `AWS-RunPatchBaseline` a commencé à s'exécuter sur un nœud géré alors qu'il est déjà en cours d'exécution dans une autre opération. Il a acquis le processus `yum` du gestionnaire de packages.

Solution : vérifiez qu'aucune association State Manager, tâche de fenêtre de maintenance ou autre configuration qui exécute `AWS-RunPatchBaseline` selon une planification ne cible le même nœud géré en même temps.

Problème : erreur indiquant « Autorisation refusée /échec d'exécution des commandes »

Problème : lorsque vous exécutez `AWS-RunPatchBaseline`, l'application des correctifs échoue avec l'erreur suivante.

```
sh:
/var/lib/amazon/ssm/instanceid/document/orchestration/commandid/PatchLinux/_script.sh:
 Permission denied
failed to run commands: exit status 126
```

Cause : `/var/lib/amazon/` peut être monté avec des autorisations `noexec`. Cela pose un problème car SSM Agent télécharge des scripts de charge utile dans `/var/lib/amazon/ssm` et les exécute depuis cet emplacement.

Solution : la vérification de la configuration des partitions exclusives est nécessaire pour `/var/log/amazon` et `/var/lib/amazon`, ainsi que leur montée avec des autorisations `exec`.

Problème : erreur indiquant « Impossible de télécharger la charge utile »

Problème : lorsque vous exécutez `AWS-RunPatchBaseline`, l'application des correctifs échoue avec l'erreur suivante.

```
Unable to download payload: https://s3.DOC-EXAMPLE-BUCKET.region.amazonaws.com/
aws-ssm-region/patchbaselineoperations/linux/payloads/patch-baseline-operations-
X.XX.tar.gz.failed to run commands: exit status 156
```

Cause : le nœud géré ne dispose pas des autorisations requises pour accéder au compartiment Amazon Simple Storage Service (Amazon S3) spécifié.

Solution : mettez à jour votre configuration réseau pour que les points de terminaison S3 soient accessibles. Pour plus de détails, consultez les informations relatives à l'accès requis aux compartiments S3 pour Patch Manager dans Communications de l'SSM Agent avec des compartiments S3 gérés par AWS.

Problème : erreur indiquant « gestionnaire de packages et combinaison de versions python non pris en charge »

Problème : lorsque vous exécutez `AWS-RunPatchBaseline`, l'application des correctifs échoue avec l'erreur suivante.

```
An unsupported package manager and python version combination was found. Apt requires
 Python3 to be installed.
failed to run commands: exit status 1
```

Cause : aucune version prise en charge de python3 n'est pas installée sur l'instance Debian Server, Raspberry Pi OS ou Ubuntu Server.

Solution : installez une version prise en charge de python3 (3.0 à 3.10) sur le serveur, comme l'exigent les nœuds gérés Debian Server, Raspberry Pi OS et Ubuntu Server.

Problème : Patch Manager n'applique pas les règles spécifiées pour exclure certains packages

Problème : vous avez tenté d'exclure certains packages en les spécifiant dans le fichier `/etc/yum.conf`, au format `exclude=`*`package-name`*, mais lors de l'opération `Install` de Patch Manager il s'avère qu'ils ne sont pas exclus.

Cause : Patch Manager n'incorpore pas les exclusions spécifiées dans le fichier `/etc/yum.conf`.

Solution : pour exclure des packages spécifiques, créez un référentiel de correctifs personnalisé et créez une règle pour exclure les packages que vous ne voulez pas installer.

Problème : l'application des correctifs échoue et Patch Manager signale que l'extension Indication de nom de serveur pour TLS n'est pas disponible

Problème : l'opération d'application de correctifs émet le message suivant.

```
/var/log/amazon/ssm/patch-baseline-operations/urllib3/util/ssl_.py:369:
SNIMissingWarning: An HTTPS request has been made, but the SNI (Server Name Indication)
 extension
to TLS is not available on this platform. This might cause the server to present an
 incorrect TLS
certificate, which can cause validation failures. You can upgrade to a newer version of
 Python
to solve this.
For more information, see https://urllib3.readthedocs.io/en/latest/advanced-
usage.html#ssl-warnings
```

Cause : ce message n'indique pas une erreur. Il s'agit plutôt d'un avertissement selon lequel l'ancienne version de Python distribuée avec le système d'exploitation ne prend pas en charge l'indication de nom de serveur TLS. Le script de charge utile du correctif Systems Manager émet cet avertissement lors de la connexion à AWS des API compatibles SNI.

Solution : pour résoudre les échecs d'application de correctifs lorsque ce message est signalé, consultez le contenu des fichiers `stdout` et `stderr`. Si vous n'avez pas configuré la ligne de base de correctifs pour stocker ces fichiers dans un compartiment S3 ou dans Amazon CloudWatch Logs, vous pouvez les localiser à l'emplacement suivant sur votre nœud géré Linux.

`/var/lib/amazon/ssm/`*`instance-id`*`/document/orchestration/`*`Run-Command-execution-id`*`/awsrunShellScript/PatchLinux`

Problème : Patch Manager signale « Plus de miroirs à essayer »

Problème : l'opération d'application de correctifs émet le message suivant.

```
[Errno 256] No more mirrors to try.
```

Cause : les référentiels configurés sur le nœud géré ne fonctionnent pas correctement. Les causes possibles incluent :

- Le cache `yum` est corrompu.
- Des problèmes liés au réseau empêchent d'atteindre une URL de référentiel.

Solution : Patch Manager utilise le gestionnaire de packages par défaut du nœud géré pour appliquer les correctifs. Vérifiez que les référentiels sont configurés et fonctionnent correctement.

Problème : le correctif échoue avec « Code d'erreur 23 renvoyé par curl »

Problème : une opération d'application de correctifs qui utilise `AWS-RunPatchBaseline` échoue avec une erreur semblable à celle-ci :

```
05/01/2023 17:04:30 root [ERROR]: Error code returned from curl is 23
```

Cause : l'outil curl utilisé sur vos systèmes ne dispose pas des autorisations nécessaires pour écrire sur le système de fichiers. Cela peut se produire lorsque l'outil curl par défaut du gestionnaire de packages a été remplacé par une version différente, telle que celle installée avec snap.

Solution : si la version curl fournie par le gestionnaire de packages a été désinstallée lors de l'installation d'une autre version, réinstallez-la.

Si vous devez conserver plusieurs versions de curl installées, assurez-vous que la version associée au gestionnaire de packages se trouve dans le premier répertoire répertorié dans la variable `PATH`. Vous pouvez le vérifier en exécutant la commande `echo $PATH` pour voir l'ordre actuel des répertoires dans lesquels les fichiers exécutables sont vérifiés sur votre système.

Problème : le correctif échoue avec le message « Error unpacking rpm package… » (Erreur de décompactage du package rpm…)

Problème : une opération de correctif échoue avec un message d'erreur similaire au suivant :

```
Error : Error unpacking rpm package python-urllib3-1.25.9-1.amzn2.0.2.noarch
python-urllib3-1.25.9-1.amzn2.0.1.noarch was supposed to be removed but is not!
failed to run commands: exit status 1
```

Cause 1 : lorsqu'un package particulier est présent dans plusieurs installateurs de packages, comme `pip` et `yum` ou `dnf`, des conflits peuvent survenir lors de l'utilisation du gestionnaire de packages par défaut.

Un exemple courant est celui du package `urllib3`, qui se trouve dans `pip`, `yum` et `dnf`.

Cause 2 : le package `python-urllib3` est endommagé. Cela peut se produire si les fichiers du package ont été installés ou mis à jour par `pip` après que le package `rpm` ait été précédemment installé par `yum` ou `dnf`.

Solution : supprimez le package `python-urllib3` de pip en exécutant la commande `sudo pip uninstall urllib3`, en conservant le package uniquement dans le gestionnaire de package par défaut (`yum` ou `dnf`).

Problème : le correctif échoue avec le message « Errors were encountered while downloading packages » (Des erreurs ont été rencontrées lors du téléchargement des packages)

Problème : pendant le correctif, vous recevez un message d'erreur similaire au suivant :

```
YumDownloadError: [u'Errors were encountered while downloading
packages.', u'libxml2-2.9.1-6.el7_9.6.x86_64: [Errno 5] [Errno 12]
Cannot allocate memory', u'libxslt-1.1.28-6.el7.x86_64: [Errno 5]
[Errno 12] Cannot allocate memory', u'libcroco-0.6.12-6.el7_9.x86_64:
```

```
[Errno 5] [Errno 12] Cannot allocate memory', u'openldap-2.4.44-25.el7_9.x86_64:
[Errno 5] [Errno 12] Cannot allocate memory',
```

Cause : cette erreur peut se produire lorsque la mémoire disponible sur un nœud géré est insuffisante.

Solution : configurez la mémoire d'échange ou mettez à niveau l'instance vers un type différent pour augmenter la prise en charge de la mémoire. Lancez ensuite une nouvelle opération de correctif.

Problème : le correctif échoue avec le message suivant : « The following signatures couldn't be verified because the public key is not available » (Les signatures suivantes n'ont pas pu être vérifiées, car la clé publique n'est pas disponible)

Problème : le correctif échoue sur Ubuntu Server avec une erreur similaire à la suivante :

```
02/17/2022 21:08:43 root [ERROR]: W:GPG error:
http://repo.mysql.com/apt/ubuntu  bionic InRelease: The following
signatures couldn't be verified because the public key is not available:
NO_PUBKEY 467B942D3A79BD29, E:The repository ' http://repo.mysql.com/apt/ubuntu bionic
```

Cause : la clé GNU Privacy Guard (GPG) a expiré ou est manquante.

Solution : actualisez la clé GPG ou ajoutez-la de nouveau.

Par exemple, en utilisant l'erreur montrée précédemment, nous voyons que la clé `467B942D3A79BD29` est manquante et doit être ajoutée. Pour ce faire, exécutez l'une des commandes suivantes :

```
sudo apt-key adv --keyserver hkps://keyserver.ubuntu.com --recv-keys 467B942D3A79BD29
```

```
sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv-keys 467B942D3A79BD29
```

Ou, pour actualiser toutes les clés, exécutez la commande suivante :

```
sudo apt-key adv --keyserver hkps://keyserver.ubuntu.com --refresh-keys
```

Si l'erreur se reproduit, nous vous recommandons de signaler le problème à l'organisation qui gère le référentiel. Jusqu'à ce qu'un correctif soit disponible, vous pouvez modifier le fichier `/etc/apt/sources.list` afin d'omettre le référentiel pendant le processus de correctif.

Pour ce faire, ouvrez le fichier `sources.list` pour le modifier, localisez la ligne relative au référentiel et insérez un caractère # au début de la ligne pour la mettre en commentaire. Ensuite, enregistrez et fermez le fichier.

Problème : l'application de correctifs échoue avec un message « NoMoreMirrorsRepoError »

Problème : vous recevez une erreur similaire à la suivante :

```
NoMoreMirrorsRepoError: failure: repodata/repomd.xml from pgdg94: [Errno 256] No more
 mirrors to try.
```

Cause : il y a une erreur dans le référentiel source.

Solution : nous vous recommandons de signaler le problème à l'organisation qui gère le référentiel. Jusqu'à ce que l'erreur soit corrigée, vous pouvez désactiver le référentiel au niveau du système d'exploitation. Pour ce faire, exécutez la commande suivante, en remplaçant la valeur de *repo-name* par le nom de votre référentiel :

```
yum-config-manager --disable repo-name
```

Voici un exemple.

```
yum-config-manager --disable pgdg94
```

Après avoir exécuté cette commande, lancez une autre opération de correctif.

Problème : le correctif échoue avec un message « Unable to download payload » (Impossible de télécharger la charge utile)

Problème : vous recevez une erreur similaire à la suivante :

```
Unable to download payload:
https://s3.dualstack.eu-west-1.amazonaws.com/aws-ssm-eu-west-1/patchbaselineoperations/
linux/payloads/patch-baseline-operations-1.83.tar.gz.
failed  to run commands: exit status 156
```

Cause : la configuration du nœud géré contient des erreurs ou est incomplète.

Solution : assurez-vous que le nœud géré est configuré avec les éléments suivants :

- Règle TCP 443 sortante dans le groupe de sécurité.
- Règle TCP 443 de sortie dans NACL.
- Règle TCP 1024-65535 d'entrée dans NACL.
- NAT/IGW dans la table de routage pour fournir une connectivité à un point de terminaison S3. Si l'instance n'a pas d'accès internet, fournissez-lui une connectivité avec le point de terminaison S3. Pour ce faire, ajoutez un point de terminaison de passerelle S3 dans le VPC et intégrez-le à la table de routage du nœud géré.

Problème : le correctif échoue avec un message « install errors: dpkg: error: dpkg frontend is locked by another process » (erreurs d'installation : dpkg : erreur : dpkg frontend est bloqué par un autre processus)

Problème : le correctif échoue avec une erreur similaire à la suivante :

```
install errors: dpkg: error: dpkg frontend is locked by another process
failed to run commands: exit status 2
Failed to install package; install status Failed
```

Cause : le gestionnaire de packages exécute déjà un autre processus sur un nœud géré au niveau du système d'exploitation. Si cet autre processus prend beaucoup de temps à se terminer, l'opération de correctif Patch Manager peut prendre du temps et échouer.

Solution : une fois que l'autre processus utilisant le gestionnaire de packages est terminé, exécutez une nouvelle opération de correctif.

Problème : le correctif sur Ubuntu Server échoue avec une erreur « dpkg was interrupted » (dpkg a été interrompu)

Problème : sur Ubuntu Server, le correctif échoue avec une erreur similaire à la suivante :

```
E: dpkg was interrupted, you must manually run
'dpkg --configure -a' to correct the problem.
```

Cause : un ou plusieurs packages sont mal configurés.

Solution : procédez comme suit :

1. Vérifiez quels sont les packages concernés et quels sont les problèmes liés à chaque package en exécutant les commandes suivantes, une à la fois :

```
sudo apt-get check
```

```
sudo dpkg -C
```

```
dpkg-query -W -f='${db:Status-Abbrev} ${binary:Package}\n' | grep -E ^.[^nci]
```

2. Corrigez les packages concernés en exécutant la commande suivante :

```
sudo dpkg --configure -a
```

3. Si la commande précédente n'a pas permis de résoudre complètement le problème, exécutez la commande suivante :

```
sudo apt --fix-broken install
```

Problème : l'utilitaire du gestionnaire de packages ne peut pas résoudre la dépendance d'un package

Problème : le gestionnaire de packages natif du nœud géré ne parvient pas à résoudre une dépendance de package et le correctif échoue. L'exemple de message d'erreur suivant indique ce type d'échec sur un système d'exploitation qui utilise `yum` comme gestionnaire de packages.

```
09/22/2020 08:56:09 root [ERROR]: yum update failed with result code: 1,
message: [u'rpm-python-4.11.3-25.amzn2.0.3.x86_64 requires rpm = 4.11.3-25.amzn2.0.3',
u'awscli-1.18.107-1.amzn2.0.1.noarch requires python2-botocore = 1.17.31']
```

Cause : sur les systèmes d'exploitation Linux, Patch Manager utilise le gestionnaire de packages natif de l'ordinateur pour exécuter les opérations de correctif. telles que `yum`, `dnf`, `apt` et `zypper`. Les applications détectent, installent, mettent à jour ou suppriment automatiquement les packages dépendants selon les besoins. Cependant, dans certaines conditions, le gestionnaire de packages peut être dans l'incapacité de mener à bien une opération de dépendance, comme par exemple :

- Plusieurs référentiels contradictoires sont configurés sur le système d'exploitation.
- L'URL d'un référentiel distant est inaccessible en raison de problèmes liés au réseau.
- Un package pour la mauvaise architecture est trouvé dans le référentiel.

Solution : le correctif peut échouer en raison d'un problème de dépendance pour une grande variété de raisons. Par conséquent, nous vous recommandons de nous contacter AWS Support pour obtenir de l'aide pour le dépannage.

Erreurs lors de l'exécution de **`AWS-RunPatchBaseline`** sur Windows Server

Rubriques

Problème : familles de produits/paires de produits dépareillées

Problèmes : lorsque vous créez un référentiel de correctifs dans la console Systems Manager, vous spécifiez une famille de produits et un produit. Par exemple, vous pouvez choisir ce qui suit :

- Famille de produits : `Office`

  Produit : `Office 2016`

Cause : si vous essayez de créer un référentiel 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 et 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 des options obsolètes ou incompatibles du produit peuvent avoir été saisis par erreur via un SDK ou une commande AWS Command Line Interface ()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 un référentiel de correctifs précédente mais qu'aucun correctif n'est disponible à partir de Microsoft.

Solution : pour éviter ce problème dans la console, sélectionnez toujours des options des sous-listes Currently available options (Options actuellement disponibles).

Vous pouvez également consulter les produits pour lesquels des correctifs sont disponibles à l'aide de la commande `describe-patch-properties` dans l' AWS CLI ou de la commande d'API `DescribePatchProperties`.

Problème: **`AWS-RunPatchBaseline`** renvoie un **`HRESULT`** (Windows Server)

Problème : vous avez reçu une erreur similaire à la suivante.

```
----------ERROR-------
Invoke-PatchBaselineOperation : Exception Details: An error occurred when
attempting to search Windows Update.
Exception Level 1:
 Error Message: Exception from HRESULT: 0x80240437
 Stack Trace: at WUApiLib.IUpdateSearcher.Search(String criteria)..
(Windows updates)
11/22/2020 09:17:30 UTC | Info | Searching for Windows Updates.
11/22/2020 09:18:59 UTC | Error | Searching for updates resulted in error: Exception
 from HRESULT: 0x80240437
----------ERROR-------
failed to run commands: exit status 4294967295
```

Cause : cette sortie indique que les API Windows Update natives n'ont pas été en mesure d'exécuter les opérations d'application de correctifs.

Solution : vérifiez le code `HResult` dans les rubriques suivantes sur microsoft.com afin d'identifier les étapes de résolution de cette erreur :

- Codes d'erreur Windows Update par composant
- Erreurs courantes et mesures d'atténuation pour Windows Update

Problème : le nœud géré n'a pas accès au catalogue Windows Update ou à WSUS

Problème : vous avez reçu une erreur similaire à la suivante.

```
Downloading PatchBaselineOperations PowerShell module from https://s3.aws-api-
domain/path_to_module.zip to C:\Windows\TEMP\Amazon.PatchBaselineOperations-1.29.zip.

Extracting PatchBaselineOperations zip file contents to temporary folder.

Verifying SHA 256 of the PatchBaselineOperations PowerShell module files.

Successfully downloaded and installed the PatchBaselineOperations PowerShell module.

Patch Summary for

PatchGroup :

BaselineId :

Baseline : null

SnapshotId :

RebootOption : RebootIfNeeded

OwnerInformation :

OperationType : Scan

OperationStartTime : 1970-01-01T00:00:00.0000000Z

OperationEndTime : 1970-01-01T00:00:00.0000000Z

InstalledCount : -1

InstalledRejectedCount : -1

InstalledPendingRebootCount : -1

InstalledOtherCount : -1

FailedCount : -1

MissingCount : -1

NotApplicableCount : -1
```

```
UnreportedNotApplicableCount : -1

EC2AMAZ-VL3099P - PatchBaselineOperations Assessment Results - 2020-12-30T20:59:46.169

----------ERROR-------

Invoke-PatchBaselineOperation : Exception Details: An error occurred when attempting to
 search Windows Update.

Exception Level 1:

Error Message: Exception from HRESULT: 0x80072EE2

Stack Trace: at WUApiLib.IUpdateSearcher.Search(String criteria)

at
 Amazon.Patch.Baseline.Operations.PatchNow.Implementations.WindowsUpdateAgent.SearchForUpdates(

searchCriteria)

At C:\ProgramData\Amazon\SSM\InstanceData\i-02573cafcfEXAMPLE\document\orchestration
\3d2d4864-04b7-4316-84fe-eafff1ea58

e3\PatchWindows\_script.ps1:230 char:13

+ $response = Invoke-PatchBaselineOperation -Operation Install -Snapsho ...

+ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

+ CategoryInfo : OperationStopped:
 (Amazon.Patch.Ba...UpdateOperation:InstallWindowsUpdateOperation) [Inv

oke-PatchBaselineOperation], Exception

+ FullyQualifiedErrorId : Exception Level 1:

Error Message: Exception Details: An error occurred when attempting to search Windows
 Update.

Exception Level 1:

Error Message: Exception from HRESULT: 0x80072EE2

Stack Trace: at WUApiLib.IUpdateSearcher.Search(String criteria)
```

```
at
 Amazon.Patch.Baseline.Operations.PatchNow.Implementations.WindowsUpdateAgent.SearchForUpdates(
 searc

---Error truncated----
```

Cause : cette erreur peut être liée aux composants Windows Update ou à un manque de connectivité au catalogue Windows Update ou aux WSUS (Windows Server Update Services).

Solution : vérifiez que le nœud géré est connecté au catalogue Microsoft Update via une passerelle Internet, une passerelle NAT ou une instance NAT. Si vous utilisez WSUS, vérifiez que le nœud géré est connecté au serveur WSUS de votre environnement. Si la connectivité est disponible pour la destination prévue, vérifiez la documentation Microsoft pour trouver d'autres causes potentielles à `HResult 0x80072EE2`. Cela peut indiquer un problème au niveau du système d'exploitation.

Problème : le PatchBaselineOperations PowerShell module n'est pas téléchargeable

Problème : vous avez reçu une erreur similaire à la suivante.

```
Preparing to download PatchBaselineOperations PowerShell module from S3.

Downloading PatchBaselineOperations PowerShell module from https://s3.aws-api-
domain/path_to_module.zip to C:\Windows\TEMP\Amazon.PatchBaselineOperations-1.29.zip.
----------ERROR-------

C:\ProgramData\Amazon\SSM\InstanceData\i-02573cafcfEXAMPLE\document\orchestration
\aaaaaaaa-bbbb-cccc-dddd-4f6ed6bd5514\

PatchWindows\_script.ps1 : An error occurred when executing PatchBaselineOperations:
 Unable to connect to the remote server

+ CategoryInfo : NotSpecified: (:) [Write-Error], WriteErrorException

+ FullyQualifiedErrorId : Microsoft.PowerShell.Commands.WriteErrorException,_script.ps1

failed to run commands: exit status 4294967295
```

Solution : vérifiez la connectivité du nœud géré et les autorisations d'accès à Amazon Simple Storage Service (Amazon S3). Le rôle du nœud géré AWS Identity and Access Management (IAM) doit utiliser les autorisations minimales citées dansCommunications de l'SSM Agent avec des compartiments S3 gérés par AWS. Le nœud doit communiquer avec le point de terminaison Amazon S3 via le

point de terminaison de la passerelle Amazon S3, la passerelle NAT ou la passerelle Internet. Pour plus d'informations sur les exigences relatives aux points de terminaison VPC pour AWS Systems Manager SSM Agent (SSM Agent), consultez. Améliorez la sécurité des instances EC2 en utilisant des points de terminaison VPC pour Systems Manager

Problème : correctifs manquants

Problème : `AWS-RunPatchbaseline` s'est terminé avec succès, mais il manque certains correctifs.

Voici quelques causes courantes et leurs solutions.

Cause 1 : le référentiel n'est pas en vigueur.

Solution 1 : pour vérifier si la cause est bien liée à ce problème, procédez comme suit :

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Run Command.
3. Sélectionnez l'onglet Historique des commandes, puis sélectionnez la commande dont vous souhaitez vérifier le référentiel.
4. Sélectionnez le nœud géré auquel il manque des correctifs.
5. Sélectionnez Étape 1 - Sortie, et recherchez la valeur `BaselineId`.
6. Vérifiez la Configuration de référentiel de correctifs affectée, c'est-à-dire le système d'exploitation, le nom du produit, la classification et la sévérité associés au référentiel de correctifs.
7. Accédez au Catalogue des mises à jour Microsoft.
8. Recherchez les ID d'articles de la base de connaissances (KB) Microsoft (par exemple, KB3216916).
9. Vérifiez que la valeur indiquée sous Product (Produit) correspond à celle de votre nœud géré, et sélectionnez le Title (Titre) correspondant. Une nouvelle fenêtre Actualiser les détails s'ouvre.
10. Sous l'onglet Présentation, la classification et la sévérité du MSRC doivent correspondre à la configuration du référentiel de correctifs que vous avez trouvée précédemment.

Cause 2 : le correctif a été remplacé.

Solution 2 : pour vérifier si cela est vrai, procédez comme suit.

1. Accédez au Catalogue des mises à jour Microsoft.

2. Recherchez les ID d'articles de la base de connaissances (KB) Microsoft (par exemple, KB3216916).
3. Vérifiez que la valeur indiquée sous Product (Produit) correspond à celle de votre nœud géré, et sélectionnez le Title (Titre) correspondant. Une nouvelle fenêtre Actualiser les détails s'ouvre.
4. Accédez à l'onglet Détails du package. Recherchez une entrée sous l'en-tête Cette mise à jour a été remplacée par les mises à jour suivantes : .

Cause 3 : le même correctif peut avoir différents numéros de KB car les mises à jour en ligne des WSUS et de Window sont gérées comme des canaux de publication indépendants par Microsoft.

Solution 3 : vérifiez l'éligibilité du correctif. Si le package n'est pas disponible sous les WSUS, installez la version 14393.3115 du système d'exploitation. Si le package est disponible pour toutes les versions du système d'exploitation, installez les versions de système d'exploitation 18362.1256 et 18363.1256.

Contacter AWS Support

Si vous ne trouvez pas de solutions de dépannage dans cette section ou dans les problèmes Systems Manager de AWS re:Post, et que vous disposez d'une formule AWS Support Développeur, Business ou Entreprise, vous pouvez formuler une demande de prise en charge technique à l'adresse AWS Support.

Avant de nous contacter AWS Support, collectez les objets suivants :

- Journaux SSM Agent
- Run CommandID de commande, ID de fenêtre de maintenance ou ID d'exécution d'Automation
- Pour les nœuds gérés Windows Server, munissez-vous également les éléments suivants :
  - `%PROGRAMDATA%\Amazon\PatchBaselineOperations\Logs` tels qu'ils figurent sous l'onglet Windows de Installation des correctifs
  - Journaux de mise à jour Windows : pour Windows Server 2012 R2 et versions antérieures, utilisez `%windir%/WindowsUpdate.log`. Pour Windows Server 2016 et les versions ultérieures, exécutez d'abord la PowerShell commande `Get-WindowsUpdateLog`avant d'utiliser `%windir%/WindowsUpdate.log`
- Pour les nœuds gérés Linux, munissez-vous également les éléments suivants :
  - Le contenu du répertoire `/var/lib/amazon/ssm/`*`instance-id`*`/document/orchestration/`*`Run-Command-execution-id`*`/awsrunShellScript/PatchLinux`.

AWS Systems Manager Distributor

Distributor, une fonctionnalité de AWS Systems Manager, vous aide à empaqueter et à publier des logiciels sur des nœuds AWS Systems Manager gérés. Vous pouvez empaqueter et publier votre propre logiciel ou l'utiliser Distributor pour rechercher et publier des packages logiciels d'agent AWS fournis AmazonCloudWatchAgent, tels que des packages tiers tels que Trend Micro.La publication d'un package annonce des versions spécifiques du document du package aux nœuds gérés que vous identifiez à l'aide d'ID de nœud, d' Compte AWS ID, de balises ou d'un Région AWS. Pour vos premiers pas dans Distributor, ouvrez Systems Manager console. Dans le panneau de navigation, sélectionnez Distributor.

Après avoir créé un package dans Distributor, vous pouvez l'installer de l'une des manières suivantes :

- Une seule fois en utilisant AWS Systems Manager Run Command
- Selon une planification en utilisant AWS Systems Manager State Manager

> **Important**
>
> Les packages distribués par des vendeurs tiers ne sont pas gérés par le fournisseur du package AWS et sont publiés par celui-ci. Nous vous encourageons à faire preuve d'une diligence raisonnable supplémentaire pour garantir le respect de vos contrôles de sécurité internes. La sécurité est une responsabilité partagée entre vous AWS et vous. Cela est décrit comme un 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

  Lorsque vous créez un package dans Distributor, le système crée un document AWS Systems Manager (document SSM). Vous pouvez joindre des fichiers .zip à ce document. Lorsque vous exécutez Distributor, le système traite les instructions du document SSM et installe le package logiciel contenu dans le fichier .zip sur les cibles spécifiées. Distributor prend en charge différents

systèmes d'exploitation, comme Windows, Ubuntu Server, Debian Server et 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.

- Contrôle de l'accès au package à partir de groupes d'instances gérées

  Vous pouvez utiliser Run Command ou State Manager pour désigner les nœuds gérés qui doivent bénéficier d'un package ainsi que la version de ce package. Run Command et State Manager sont des fonctionnalités d' AWS Systems Manager. Les nœuds gérés peuvent être regroupés par ID d'instance ou d'appareil, par Compte AWS numéro, par étiquette ou Régions AWS. Vous pouvez utiliser des associations State Manager pour diffuser différentes versions d'un package à différents groupes d'instances.

- De nombreux packages d' AWS agents inclus et prêts à l'emploi

  Distributorinclut de nombreux packages d' AWS agents prêts à être déployés sur des nœuds gérés. Sur la page de la liste `Packages` Distributor, recherchez les packages publiés par `Amazon`. Exemples : `AmazonCloudWatchAgent` et `AWSPVDriver`.

- Automatisation du déploiement

  Pour maintenir votre environnement à jour, utilisez State Manager afin de planifier le déploiement automatique des packages sur les nœuds gérés cibles lors du premier lancement de ces derniers.

À qui est destiné Distributor ?

- Tout AWS client souhaitant créer de nouveaux packages logiciels ou déployer des packages logiciels existants, y compris des packages AWS publiés, sur plusieurs nœuds gérés par Systems Manager à la fois.
- Les développeurs de logiciels qui créent des packages logiciels.
- Administrateurs chargés de maintenir les nœuds gérés par Systems Manager à jour avec la plupart des packages up-to-date logiciels.

Quelles sont les fonctions d'Distributor ?

- Déploiement de packages sur les instances Windows et Linux

  AvecDistributor, vous pouvez déployer des packages logiciels sur des instances Amazon Elastic Compute Cloud (Amazon EC2) et des appareils principaux pour AWS IoT Greengrass 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".

> **Note**
>
> Distributor n'est pas pris en charge sur le système d'exploitation macOS.

- 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 fonctionnalités de la console PowerShell, de la CLI et du SDK Distributor

  Vous pouvez travailler avec en Distributor utilisant la console Systems Manager AWS Command Line Interface (AWS CLI) ou le AWS SDK de votre choix. AWS Tools for PowerShell

- Contrôle d'accès IAM

  En utilisant des politiques AWS Identity and Access Management (IAM), vous pouvez contrôler quels membres de votre organisation peuvent créer, mettre à jour, déployer ou supprimer des packages ou des versions de packages. 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 auditer et enregistrer les actions des Distributor utilisateurs dans votre compte Compte AWS grâce à l'intégration avec d'autres Services AWS. Pour de plus amples informations, veuillez consulter Audit et journalisation de l'activité de Distributor.

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. Windows Serverles nœuds gérés basés sur la PowerShell technologie nécessitent des scripts (scripts nommés `install.ps1` et`uninstall.ps1`). Les nœuds gérés basés sur Linux nécessitent des scripts shell (scripts nommés `install.sh` et). `uninstall.sh` AWS Systems Manager SSM Agentlit et exécute les instructions contenues dans les uninstall scripts install et.
  - Un fichier exécutable. SSM Agent doit trouver ce fichier exécutable pour installer le package sur les nœuds gérés cibles.
- 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 Simple Storage Service (Amazon S3) que les fichiers .zip qui constituent le package. Le manifeste identifie la version du package et mappe les fichiers .zip du package sur les attributs du nœud géré 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 : Création du manifeste de package JSON.

Lorsque vous sélectionnez 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

Vous pouvez utiliser Distributor pour publier des packages sur les plateformes de nœuds gérés Systems Manager suivantes. Une valeur de version doit correspondre à la version exacte de l'Amazon Machine Image (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 : Création du manifeste de package JSON.

> **Note**
>
> Systems Manager ne prend pas en charge tous les systèmes d'exploitation suivants pour les appareils AWS IoT Greengrass principaux. Pour plus d'informations, consultez la section Configuration des appareils AWS IoT Greengrass principaux dans le Guide du AWS IoT Greengrass Version 2 développeur.

Plateforme	Valeur de code dans le fichier manifeste	Architecture
Windows Server	`windows`	`x86_64` ou `386`
Debian Server	`debian`	`x86_64` ou `386`
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 1, Amazon Linux 2 et Amazon Linux 2023	`amazon`	`x86_64` ou `386` `arm64` (Amazon Linux 2 et AL2023, types d'instance A1)
SUSE Linux Enterprise Server (SLES)	`suse`	`x86_64` ou `386`
openSUSE	`opensuse`	`x86_64` ou `386`

Plateforme	Valeur de code dans le fichier manifeste	Architecture
openSUSE Leap	`opensuseleap`	`x86_64` ou `386`
Oracle Linux	`oracle`	`x86_64`

Rubriques

Configuration de Distributor

Avant d'utiliser Distributor, une fonctionnalité de AWS Systems Manager, pour créer, gérer et déployer des packages logiciels, exécutez les étapes ci-après.

Rubriques

Étape 1 : Exécution des conditions Distributor prérequises

Avant d'utiliser Distributor, une fonctionnalité de AWS Systems Manager, vérifiez que votre environnement respecte les conditions requises suivantes.

Conditions préalables requises Distributor

Exigence	Description
SSM Agent	AWS Systems Manager SSM Agent 2.3.274.0 (ou version ultérieure) doit être installé sur

Exigence	Description
	les nœuds gérés sur lesquels vous souhaitez déployer ou supprimer des packages. Pour installer ou mettre à jour SSM Agent, consultez Utilisation de l'option SSM Agent.
AWS CLI	(Facultatif) Pour utiliser l'AWS Command Line Interface (AWS CLI) à la place de la console 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'install ation ou la mise à niveau de la CLI, veuillez consulter Installation de la AWS Command Line Interfacedans le Guide de l'utilisateur AWS Command Line Interface.
AWS Tools for PowerShell	(Facultatif) Pour utiliser les Tools for Powershel l à la place de la console Systems Manager pour créer et gérer des packages, installez la dernière version des Tools for Powershell sur votre ordinateur local. Pour plus d'informations sur l'installation ou la mise à niveau de Tools for PowerShell, reportez-vous à Installation de AWS Tools for Windows PowerShell ou à AWS Tools for PowerShell Core dans le Guide de l'utilisateur de AWS Tools for Windows PowerShell.

> **Note**
>
> Systems Manager ne prend pas en charge la distribution de packages aux nœuds gérés Oracle Linux à l'aide de Distributor.

Étape 2 : Vérification ou création d'un profil d'instance IAM avec les autorisations Distributor

Par défaut, AWS Systems Manager n'est pas autorisé à effectuer des actions sur vos instances. Vous devez accorder l'accès à l'aide d'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 pour toutes les fonctionnalités de Systems Manager, et pas seulement Distributor pour celles qui sont des fonctionnalités de AWS Systems Manager.

> **Note**
>
> Lorsque vous configurez vos périphériques périphériques pour exécuter le logiciel AWS IoT Greengrass CoreSSM Agent, vous spécifiez un rôle de service IAM qui permet à Systems Manager d'effectuer des actions sur celui-ci. Il n'est pas nécessaire de configurer les appareils de périphérie gérés avec un profil d'instance.

Si vous utilisez déjà d'autres fonctionnalités Systems Manager, par exemple, Run Command et State Manager, un profil d'instance avec les autorisations requises pour Distributor est déjà attaché à vos instances. Le moyen le plus simple de vous assurer que vous êtes autorisé à effectuer des Distributor tâches consiste à associer la politique ManagedInstanceprincipale d'AmazonSSM à votre profil d'instance. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager.

Étape 3 : Contrôle de l'accès utilisateur aux packages

Les politiques AWS Identity and Access Management (IAM) vous permettent de contrôler les personnes pouvant créer, déployer et gérer des packages. Vous pouvez également contrôler les opérations d'API Run Command et State Manager que les utilisateurs sont autorisés à exécuter sur les nœuds gérés. À l'instar de Distributor, Run Command et State Manager sont des fonctionnalités de AWS Systems Manager.

Format ARN

Les packages définis par l'utilisateur sont associés à des Amazon Resource Names (ARN) de document et ont le format suivant.

```
arn:aws:ssm:region:account-id:document/document-name
```

Voici un exemple.

```
arn:aws:ssm:us-west-1:123456789012:document/ExampleDocumentName
```

Vous pouvez utiliser une paire de politiques 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 politiques IAM personnalisées appropriées, répondant à vos exigences en matière d'autorisations.

Pour de plus amples informations sur l'utilisation de variables dans les politiques IAM, consultez Éléments de politique IAM : Variables.

Pour obtenir des informations sur la création de politiques et la façon de les attacher à des utilisateurs ou des groupes, consultez Création de politiques IAM et Ajout et suppression de politiques IAM dans le Guide de l'utilisateur IAM.

Étape 4 : créer ou choisir un compartiment Amazon S3

Lorsque vous créez un package en utilisant le flux de travail Simple dans la console AWS Systems Manager, vous sélectionnez un compartiment Amazon Simple Storage Service (Amazon S3) existant dans lequel Distributor charge votre logiciel. Distributor est une fonctionnalité de AWS Systems Manager. Dans le flux de travail Advanced (Avancé), vous devez charger les fichiers .zip de vos logiciels ou ressources dans un compartiment Amazon 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 Amazon 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 Systems Manager interne. Étant donné que les ressources sont copiées vers un magasin interne, vous pouvez supprimer ou réutiliser votre compartiment Amazon S3 lors de la création du package est terminée.

Pour plus d'informations sur la création d'un compartiment, veuillez consulter 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 console AWS Systems Manager, les outils de ligne de commande AWS (AWS CLI et AWS Tools for PowerShell) et les kits SDK AWS pour ajouter, gérer ou déployer des packages

dans Distributor. Distributor est une fonctionnalité de AWS Systems Manager. Avant d'ajouter un package à Distributor :

- Créez et zippez 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.
- Prévoyez un compartiment Amazon Simple Storage Service (Amazon S3) pour stocker vos ressources ou logiciels installables. Si vous utilisez le processus de création de package Advanced (Avancé), chargez vos ressources dans le compartiment Amazon 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 nœuds gérés, consultez Installer ou mettre à jour des packages.

Vous pouvez partager des packages Distributor entre des Comptes AWS. Lors de l'utilisation d'un package partagé à partir d'un autre compte dans des commandes AWS CLI, utilisez l'Amazon Resource Name (ARN) du package plutôt que son nom.

Rubriques

- Supprimer un package

Afficher les packages

Pour afficher les packages disponibles à l'installation, vous pouvez utiliser la console AWS Systems Manager ou votre outil de ligne de commande AWS préféré. Distributor est une fonctionnalité de AWS Systems Manager. Pour accéder à Distributor, ouvrez la console AWS Systems Manager et sélectionnez Distributor dans le panneau de navigation de gauche. Vous verrez s'afficher tous les forfaits disponibles.

La section suivante décrit l'affichage de packages Distributor en utilisant votre outil de ligne de commande préféré.

Afficher les packages (ligne de commande)

Cette section contient des informations sur l'utilisation de votre outil de ligne de commande préféré pour afficher des packages Distributor à l'aide des commandes fournies.

Linux & macOS

Pour afficher des packages à l'aide de la AWS CLI sous Linux

- Pour afficher tous les packages, à l'exclusion des packages partagés, exécutez la commande suivante.

```
aws ssm list-documents \
    --filters Key=DocumentType,Values=Package
```

- Pour afficher tous les packages appartenant à Amazon, exécutez la commande suivante.

```
aws ssm list-documents \
    --filters Key=DocumentType,Values=Package Key=Owner,Values=Amazon
```

- Pour afficher tous les packages appartenant à des tiers, exécutez la commande suivante.

```
aws ssm list-documents \
    --filters Key=DocumentType,Values=Package Key=Owner,Values=ThirdParty
```

Windows

Pour afficher des packages à l'aide de la AWS CLI sous Windows

- Pour afficher tous les packages, à l'exclusion des packages partagés, exécutez la commande suivante.

```
aws ssm list-documents ^
    --filters Key=DocumentType,Values=Package
```

- Pour afficher tous les packages appartenant à Amazon, exécutez la commande suivante.

```
aws ssm list-documents ^
    --filters Key=DocumentType,Values=Package Key=Owner,Values=Amazon
```

- Pour afficher tous les packages appartenant à des tiers, exécutez la commande suivante.

```
aws ssm list-documents ^
    --filters Key=DocumentType,Values=Package Key=Owner,Values=ThirdParty
```

PowerShell

Pour afficher des packages à l'aide de Tools for PowerShell

- Pour afficher tous les packages, à l'exclusion 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 appartenant à 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 appartenant à 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.

Lorsque vous créez un package, le système crée un nouveau document SSM. Ce document vous permet de déployer le package sur des nœuds gérés.

À des fins de démonstration uniquement, un exemple de package, ExamplePackage.zip, peut être téléchargé sur notre site Web. Le package d'exemple inclut un manifeste JSON complet et trois

fichiers .zip contenant les programmes d'installation pour PowerShell la version 7.0.0. Les scripts d'installation et de désinstallation ne contiennent pas de commandes valides. Même si vous devez zipper 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 zippez pas les ressources installables dans le flux de travail Simple.

Rubriques

Créer un package (simple)

Cette section décrit comment créer un package dans en Distributor choisissant le flux de travail de création de package simple dans la Distributor console. Distributorest une capacité de AWS Systems Manager. 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 zip de vos fichiers installables, ainsi que la création d'un nouveau package et du document SSM associé. Pour de plus amples informations sur les plateformes prises en charge, veuillez consulter Architectures et plateformes de package prises en charge.

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**
>
> Utilisez l'option de mise à jour `In-place` pour 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 AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.

2. Dans le panneau de navigation, sélectionnez Distributor.
3. Sur la page d'accueil de Distributor, sélectionnez 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), sélectionnez un compartiment en utilisant le nom et le préfixe du compartiment ou en utilisant son URL.
7. Dans Upload software (Charger un logiciel), sélectionnez Add software (Ajouter un logiciel), puis recherchez les fichiers de logiciels installables qui portent l'extension `.rpm`, `.msi` ou `.deb`. Si le nom de fichier contient des espaces, le téléchargement échoue. 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, sélectionnez 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), zippez 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 Créer un package (avancé).
   - 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 : Création du manifeste de package JSON.

9. Pour Platform version (Version de plateforme), vérifiez que la version de plateforme de système d'exploitation est **_any**, une version majeure suivie d'un caractère générique (7.*), ou la version de système d'exploitation exacte spécifique 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 : Création du manifeste de package JSON.
10. Pour Architecture, sélectionnez 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.
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, sélectionnez 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, sélectionnez 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, sélectionnez 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 : Création du manifeste de package JSON.
15. Sélectionnez 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, sélectionnez 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, sélectionnez 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 ManagerDistributor.

Créer un package (avancé)

Dans cette section, découvrez comment les utilisateurs avancés peuvent créer un package dans Distributor après avoir chargé des ressources installables zippées avec des scripts d'installation et de désinstallation, ainsi qu'un fichier manifeste JSON dans un compartiment Amazon 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 ou un répertoire 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 sur notre site Web. Cet exemple de package comprend un manifeste JSON complet et trois fichiers .zip.

Rubriques

- Étape 1 : Création des fichiers ZIP
- Étape 2 : Création du manifeste de package JSON
- Étape 3 : Chargement du package et du manifeste dans un compartiment Amazon S3
- Étape 4 : Ajout d'un package à Distributor

Étape 1 : Création des 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. Windows Serverles nœuds gérés basés sur la PowerShell technologie nécessitent des scripts (scripts nommés `install.ps1` et`uninstall.ps1`). Les nœuds gérés basés sur Linux requièrent des scripts shell (scripts nommés `install.sh` et `uninstall.sh`). SSM Agent 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 du package hors ligne pendant la mise à jour. Windows Serverles nœuds gérés basés sur la technologie nécessitent un PowerShell script (nom du script`update.ps1`). Les nœuds gérés basés sur Linux requièrent un script shell (script nommé `update.sh`). SSM Agent 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.

Pour obtenir des exemples de fichiers .zip, y compris des exemples install et uninstall des scripts, téléchargez le package d'exemple, ExamplePackage.zip.

Étape 2 : Création du manifeste de package JSON

Une fois que vous avez préparé et zippé 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 également créer le manifeste dans la AWS Systems Manager console 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.

   ```
   "version": "1.0.1",
   ```

3. (Facultatif) Ajoutez un nom d'éditeur. Voici un exemple.

   ```
   "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 Architectures et plateformes de package prises en charge.

   L'élément *`platform-version`* peut prendre 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 pour que toutes les versions mineures soient prises en charge, par exemple 7.*. Si vous choisissez de spécifier une valeur *`platform-version`* pour une version spécifique du système d'exploitation, vérifiez 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 nœud géré basé sur Windows Server, la version est disponible sous forme de données WMI (Windows Management Instrumentation). Vous pouvez exécuter la commande suivante à partir d'une invite de commande pour obtenir la version, puis analyser les résultats obtenus pour `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 un nœud géré 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, sur Debian Server, vous pouvez analyser le fichier `/etc/debian_version`, ou sur Red Hat Enterprise Linux, le fichier `/etc/redhat-release`.

  ```
  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. 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 nœuds gérés dotés de l'architecture `x86_64`. Les versions `2016.09` et `_any` sont toutes deux des entrées situées sous `amazon`. Il y a une seule plateforme (Amazon Linux), mais différentes versions et architectures, et différents fichiers .zip associés pris en charge.

```
{
    "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 prenant en charge à la fois les versions Red Hat Enterprise Linux 7.x et Amazon Linux, vous avez deux entrées dans la section `"packages"` pointant 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 zippées 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 `shasum -a 256` *`file-name`*`.zip` ou `openssl dgst -sha256` *`file-name`*`.zip`. 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 sur notre site Web. Cet exemple de package comprend un manifeste JSON complet et trois fichiers .zip.

Étape 3 : Chargement du package et du manifeste dans un compartiment Amazon 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 : Création du manifeste de package JSON 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 zippez pas le dossier ou le répertoire dans lequel vous déplacez vos fichiers d'archive .zip et votre fichier manifeste.
2. Créez un compartiment ou sélectionnez un compartiment existant. Pour plus d'informations, veuillez consulter Créer un compartiment dans le Guide de mise en route Amazon Simple Storage Service. Pour plus d'informations sur la façon d'exécuter une AWS CLI commande pour créer un bucket, consultez mbla référence des AWS CLI commandes.
3. Téléchargez le dossier ou le répertoire dans le compartiment. Pour plus d'informations, veuillez consulter Ajouter un objet à un compartiment dans le Guide de mise en route Amazon Simple Storage Service. Si vous prévoyez de coller votre manifeste JSON dans la AWS Systems Manager console, ne le chargez pas. Pour plus d'informations sur la façon d'exécuter une AWS CLI commande pour télécharger des fichiers mvdans un bucket, consultez la référence des AWS CLI commandes.
4. Sur la page d'accueil du compartiment, sélectionnez le dossier ou le répertoire 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 : Ajout d'un package à Distributor

Vous pouvez utiliser la AWS Systems Manager console, les outils de ligne de AWS commande (AWS CLI et AWS Tools for PowerShell) ou AWS les SDK pour y ajouter un nouveau package. Distributor Lorsque vous ajoutez un package, vous ajoutez un nouveau document SSM. Ce document vous permet de déployer le package sur des nœuds gérés.

Rubriques

- Ajout d'un package (console)
- Ajout d'un package (AWS CLI)

Ajout d'un package (console)

Vous pouvez utiliser la AWS Systems Manager console pour créer un package. Vous devez avoir à votre disposition le nom du compartiment dans lequel vous avez chargé votre package dans Étape 3 : Chargement du package et du manifeste dans un compartiment Amazon S3.

Pour ajouter un package dans Distributor (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Distributor.
3. Sur la page d'accueil de Distributor, sélectionnez 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), sélectionnez le nom du compartiment dans lequel vous avez chargé vos fichiers .zip et le manifeste dans the section called "Étape 3 : Chargement du package et du manifeste dans un compartiment Amazon S3".
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), sélectionnez Extract from package (Extraire depuis le package) pour utiliser un manifeste que vous avez chargé dans le compartiment Amazon 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, sélectionnez 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 : Création du manifeste de package JSON.
9. Lorsque vous avez fini avec le fichier manifeste, sélectionnez 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, sélectionnez Cancel (Annuler).

Ajout d'un package (AWS CLI)

Vous pouvez utiliser le 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 : Chargement du package et du manifeste dans un compartiment Amazon S3.

Pour ajouter un package dans Amazon S3 (AWS CLI)

1. AWS CLI Pour créer un package, exécutez la commande suivante en remplaçant *package-name par le nom* de votre package et *path-to-manifest-file par le chemin du fichier* manifeste JSON. DOC-EXAMPLE-BUCKET est l'URL du compartiment Amazon S3 dans lequel le package complet 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 Amazon 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.

   ```
   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 de la Référence de Command AWS CLI . 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)

Après avoir ajouté un package à Distributor, une fonctionnalité de AWS Systems Manager, vous pouvez modifier les autorisations du package dans la console Systems Manager. 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.

> **Note**
>
> Vous pouvez mettre à jour les autorisations des packages qui sont partagés avec au maximum 20 comptes.

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, sélectionnez Distributor.

3. Dans la page Packages, sélectionnez le package dont vous souhaitez modifier les autorisations.
4. Sur l'onglet Package details (Détails du package), sélectionnez Edit permissions (Modifier les autorisations) pour pouvoir modifier les autorisations.
5. Pour Edit permissions (Modifier les autorisations), sélectionnez 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, sélectionnez Enregistrer.

Modifier les balises du package (console)

Après avoir ajouté un package à Distributor, une fonctionnalité de AWS Systems Manager, vous pouvez modifier les balises du package dans la console Systems Manager. Ces balises sont appliquées au package, et ne sont pas liées aux balises du nœud géré sur lequel 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, sélectionnez Distributor.
3. Dans la page Packages, sélectionnez le package dont vous souhaitez modifier les balises.
4. Sur l'onglet Package details (Détails du package), dans Tags (Balises), sélectionnez Edit (Modifier).
5. Pour Add tags (Ajouter des balises), entrez une clé de balise ou une paire clé de balise-valeur, puis sélectionnez Add (Ajouter). Répétez cette opération si vous souhaitez ajouter d'autres balises. Pour supprimer des balises, sélectionnez X sur la balise en bas de la fenêtre.
6. Une fois que vous avez fini d'ajouter les balises au package, sélectionnez Save (Enregistrer).

Ajouter une version de package à Distributor

Pour ajouter une version de package, créez un package, puis utilisez-le Distributor pour ajouter une version de package en ajoutant une entrée au document AWS Systems Manager (SSM) qui existe déjà pour les anciennes versions. Distributorest une capacité de AWS Systems Manager. 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 Amazon Simple Storage Service (Amazon S3), mais doit avoir une URL avec un autre nom de fichier affiché à la fin. Vous pouvez utiliser la console Systems Manager ou la AWS Command Line Interface (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 Amazon 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 Amazon 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)
- Ajout d'une version de package (AWS CLI)

Ajout d'une version de package (console)

Avant d'effectuer ces étapes, suivez les instructions définies dans Créer un package afin de créer un nouveau package pour la version. Utilisez ensuite la console 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 AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Distributor.
3. Sur la page d'accueil de Distributor, sélectionnez le package auquel vous voulez ajouter une autre version.
4. Sur la page Add version (Ajouter une version), sélectionnez 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), sélectionnez 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, sélectionnez 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), zippez 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 Ajout d'une version de package (avancé).
   - 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 : Création du manifeste de package JSON.

10. Pour Platform version (Version de plateforme), vérifiez que la version de plateforme de système d'exploitation est **_any**, une version majeure suivie d'un caractère générique (7.*), ou la version de système d'exploitation exacte spécifique 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 : Création du manifeste de package JSON.
11. Pour Architecture, sélectionnez 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.
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, sélectionnez 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, sélectionnez 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 : Création du manifeste de package JSON.

15. Lorsque vous avez fini d'ajouter des logiciels et de vérifier les données de plateforme cible, de version et d'architecture, sélectionnez 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, sélectionnez 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, sélectionnez 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 ManagerDistributor.
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, 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 AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Distributor.
3. Sur la page d'accueil de Distributor, sélectionnez le package auquel vous voulez ajouter une autre version, puis sélectionnez 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), sélectionnez 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), sélectionnez 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 Amazon S3 où vous avez stocké vos fichiers .zip, sélectionnez 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 : Création du manifeste de package JSON.
8. Lorsque vous avez fini avec le fichier manifeste, sélectionnez Add package version (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 le AWS CLI pour ajouter une nouvelle version de package àDistributor. 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 AWS Systems Manager document contenant une entrée pour une nouvelle version du package. Remplacez *document-name* par le nom de votre document. Remplacez *DOC-EXAMPLE-BUCKET* par l'URL du manifeste JSON que vous avez copiée dans Étape 3 : Chargement du package et du manifeste dans un compartiment Amazon S3. *S3-bucket-URL-of-package* est l'URL du compartiment Amazon S3 dans lequel le package complet est stocké. Remplacez *version-name-from-updated-manifest* par 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.

   ```
   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* par le numéro de version du document (pas le même que celui de la version de 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 plus d'informations sur les autres options que vous pouvez utiliser avec la update-document commande, reportez-vous update-documentà la AWS Systems Manager section de la référence des AWS CLI commandes.

Installer ou mettre à jour des packages

Vous pouvez déployer des packages sur vos nœuds AWS Systems Manager gérés en utilisant Distributor une fonctionnalité de AWS Systems Manager. Pour déployer les packages, utilisez le AWS Management Console ou AWS Command Line Interface (AWS CLI). Vous pouvez 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 State Manager d'utiliser une fonctionnalité de AWS Systems Manager pour installer des packages. L'utilisation State Manager permet de garantir que vos nœuds gérés exécutent toujours la up-to-date version la plus complète de votre package.

Préférence	AWS Systems Manager action	Plus d'informations
Installer ou mettre à jour un package immédiatement.	Run Command	• Installation ou mise à jour d'un package une fois (console) • Installation unique d'un package (AWS CLI) • Mise à jour unique d'un package (AWS CLI)
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)

Préférence	AWS Systems Manager action	Plus d'informations
		• Planification d'une installation de package (AWS CLI) • Planification d'une mise à jour de package (AWS CLI)
Installer automatiquement un package sur de nouveaux nœuds gérés dotés d'une balise spécifique ou d'un ensemble de balises spécifiques. Par exemple, l'installation de l' CloudWatch agent Amazon sur de nouvelles instances.	State Manager	Pour ce faire, vous pouvez appliquer des balises à de nouveaux nœuds gérés, puis spécifier les balises en tant que cibles dans votre association State Manager. State Manager installe automatiquement le package dans une association sur les nœuds gérés dotés de balises correspondantes. veuillez consulter À propos des cibles et des contrôles du débit dans les associations State Manager.

Rubriques

- Installation ou mise à jour d'un package une fois (console)
- Planification d'une installation ou d'une mise à jour de package (console)
- Installation unique d'un package (AWS CLI)
- Mise à jour unique d'un package (AWS CLI)
- Planification d'une installation de package (AWS CLI)
- Planification d'une mise à jour de package (AWS CLI)

Installation ou mise à jour d'un package une fois (console)

Vous pouvez utiliser la AWS Systems Manager console pour installer ou mettre à jour un package une seule fois. Lorsque vous configurez une installation unique, Distributor utilise AWS Systems Manager Run Command, une fonctionnalité de AWS Systems Manager, pour effectuer l'installation.

Pour installer ou mettre à jour un package une fois (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Distributor.
3. Dans la page d'accueil de Distributor, sélectionnez le package à installer.
4. Sélectionnez Installer une fois.

   Cette commande ouvre Run Command 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, sélectionnez Installer.
7. Pour Installation type (Type d'installation), sélectionnez l'une des valeurs suivantes :

   - Uninstall and reinstall (Désinstaller et réinstaller) : le package est complètement désinstallé, puis réinstallé. L'application n'est pas disponible tant que la réinstallation n'est pas terminée.
   - In-place update (Mise à jour sur place) : seuls les fichiers nouveaux ou modifiés sont ajoutés à l'installation existante, conformément aux instructions que vous fournissez dans un script `update`. L'application reste disponible tout au long du processus de mise à jour. Cette option n'est pas prise en charge pour les packages AWS publiés, à l'exception `AWSEC2Launch-Agent` du package.
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, Run Command installe la version par défaut que vous avez sélectionnée dans Distributor.
10. Dans la section Targets (Cibles), sélectionnez les nœuds gérés sur lesquels vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant des instances ou des appareils manuellement, ou en spécifiant un groupe de ressources.

> **Note**
>
> Si un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés.

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. Dans Rate Control (Contrôle de débit) :

    - Dans Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage de cibles sur lesquelles exécuter simultanément la commande.

      > **Note**
      >
      > Si vous avez sélectionné des cibles en spécifiant des balises ou des groupes de ressources et que vous n'êtes pas certain du nombre de nœuds gérés ciblés, limitez le nombre de cibles autorisées à exécuter simultanément le document en indiquant un pourcentage.

    - Dans Error threshold (Seuil d'erreur), indiquez quand arrêter l'exécution de la commande sur les autres cibles après l'échec de celle-ci sur un certain nombre ou un certain pourcentage de nœuds gérés. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les nœuds gérés sur lesquels la commande est toujours 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). Saisissez 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 (pour les instances EC2) ou de la fonction du service IAM (pour les machines activées par un système hybride) attribués

à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, assurez-vous que le profil d'instance ou le rôle de service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

14. Dans la section SNS notifications (Notifications SNS), si vous souhaitez envoyer des notifications sur le statut d'exécution des commandes, cochez la case Enable SNS notifications (Activer les notifications SNS).

    Pour plus d'informations sur la configuration des notifications Amazon SNS pour Run Command, consultez Surveillance des changements d'état du Systems Manager à l'aide des notifications Amazon SNS.
15. Lorsque vous êtes prêt à installer le package, sélectionnez Run (Exécuter).
16. La zone Command status (État de la commande) indique la progression de l'exécution. Si la commande est toujours en cours, sélectionnez 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 situé en regard d'un nom de nœud géré, puis sélectionnez 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 Amazon S3, sélectionnez 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 AWS Systems Manager console 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 pour effectuer l'installation ou la mise à jour.

Pour planifier une installation de package (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Distributor.

3. Dans la page d'accueil de Distributor, sélectionnez le package à installer ou mettre à jour.
4. Pour Package, sélectionnez Install on a schedule (Installation planifiée).

   Cette commande ouvre 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), sélectionnez l'une des valeurs suivantes :

   - Uninstall and reinstall (Désinstaller et réinstaller) : le package est complètement désinstallé, puis réinstallé. L'application n'est pas disponible tant que la réinstallation n'est pas terminée.
   - In-place update (Mise à jour sur place) : seuls les fichiers nouveaux ou modifiés sont ajoutés à l'installation existante, conformément aux instructions que vous fournissez dans un script `update`. 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), sélectionnez 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.

    > **Note**
    >
    > Vous pouvez choisir les appareils AWS IoT Greengrass principaux gérés en choisissant soit Sélection de toutes les instances gérées dans ce compte, soit Sélection manuelle de l'instance.
12. Pour Specify schedule (Spécifier le calendrier), sélectionnez 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 plus d'informations sur ces options, consultez Utilisation d'associations dans Systems Manager. Utilisez les contrôles pour créer un calendrier de type `cron` ou rate pour l'association.

13. Sélectionnez Create Association (Créer une association).
14. Sur la page Association cliquez sur le bouton en regard de l'association que vous avez créée, puis sélectionnez Apply association now (Appliquer l'association maintenant).

    State Manager crée et exécute immédiatement l'association sur les cibles spécifiées. Pour de plus amples informations sur les résultats de l'exécution des associations, consultez Utilisation d'associations dans Systems Manager 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), consultez Utilisation d'associations dans Systems Manager.

Installation unique d'un package (AWS CLI)

Vous pouvez exécuter send-command le AWS CLI pour installer un Distributor package une seule fois. 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.

```
aws ssm send-command \
```

```
    --document-name "AWS-ConfigureAWSPackage" \
    --instance-ids "i-00000000000000" \
    --parameters '{"action":["Install"],"installationType":["Uninstall and
 reinstall"],"name":["ExamplePackage"]}'
```

Pour plus d'informations sur les autres options que vous pouvez utiliser avec la send-command commande, reportez-vous send-commandà la AWS Systems Manager section de la référence des AWS CLI commandes.

Mise à jour unique d'un package (AWS CLI)

Vous pouvez l'exécuter send-command AWS CLI pour mettre à jour un Distributor package sans mettre l'application associée hors ligne. 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.

```
aws ssm send-command \
    --document-name "AWS-ConfigureAWSPackage" \
    --instance-ids "i-02573cafcfEXAMPLE" \
    --parameters '{"action":["Install"],"installationType":["In-place
 update"],"name":["ExamplePackage"]}'
```

Pour plus d'informations sur les autres options que vous pouvez utiliser avec la send-command commande, reportez-vous send-commandà la AWS Systems Manager section de la référence des AWS CLI commandes.

Planification d'une installation de package (AWS CLI)

Vous pouvez exécuter create-association le AWS CLI pour installer un Distributor package selon un calendrier. La valeur de `--name`, le nom du document, est toujours `AWS-ConfigureAWSPackage`. La commande suivante utilise la clé `InstanceIds` pour spécifier des nœuds gérés 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.

```
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 plus d'informations sur les autres options que vous pouvez utiliser avec la create-association commande, reportez-vous create-associationà la AWS Systems Manager section de la référence des AWS CLI commandes.

Planification d'une mise à jour de package (AWS CLI)

Vous pouvez exécuter create-association le AWS CLI pour mettre à jour un Distributor package selon un calendrier sans mettre l'application associée hors ligne. 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.

```
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 plus d'informations sur les autres options que vous pouvez utiliser avec la create-association commande, reportez-vous create-associationà la AWS Systems Manager section de la référence des AWS CLI commandes.

Désinstaller un package

Vous pouvez utiliser la AWS Management Console ou l'AWS Command Line Interface (AWS CLI) pour désinstaller des packages Distributor à partir de vos nœuds gérés AWS Systems Manager à l'aide de Run Command. Distributor et Run Command sont des fonctionnalités d'AWS Systems

Manager. 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)

Vous pouvez utiliser Run Command dans la console Systems Manager pour désinstaller un package ponctuellement. Distributor utilise AWS Systems Manager Run Command 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, sélectionnez Run Command.
3. Sur la page d'accueil de Run Command, sélectionnez Run command (Exécuter la commande).
4. Sélectionnez le document de commande `AWS-ConfigureAWSPackage`.
5. Dans Action, sélectionnez Uninstall (Désinstaller)
6. Pour Name (Nom), tapez le nom du package à désinstaller.
7. Dans Targets (Cibles), sélectionnez la façon dont vous souhaitez cibler vos nœuds gérés. Vous pouvez spécifier une clé et des valeurs de balise partagées par les cibles. Vous pouvez également spécifier des cibles en choisissant des attributs, comme un ID, une plateforme et une version de SSM Agent.
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 Simple Notification Service (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, sélectionnez Run (Exécuter), puis View results (Afficher les résultats).

10. Dans la liste des commandes, sélectionnez la commande `AWS-ConfigureAWSPackage` que vous venez d'exécuter. Si la commande est toujours en cours d'exécution, sélectionnez l'icône d'actualisation dans le coin supérieur droit de la console.
11. Lorsque la colonne Status (Statut) affiche Success (Réussite) ou Failed (Échec), sélectionnez l'onglet Output (Sortie).
12. Sélectionnez 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 la AWS CLI pour désinstaller un package Distributor de vos nœuds gérés à l'aide de Run Command.

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.

```
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, consultez send-command dans la section AWS Systems Manager de la Référence de Command AWS CLI.

Supprimer un package

Cette section explique comment supprimer un package. Vous ne pouvez pas supprimer une version particulière d'un package, seulement le package entier.

Suppression d'un package (console)

Vous pouvez utiliser la console AWS Systems Manager pour supprimer un package ou une version de package à partir de Distributor, une fonctionnalité de AWS Systems Manager. 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, sélectionnez Distributor.
3. Dans la page d'accueil de Distributor, sélectionnez le package à supprimer.
4. Sur la page des détails du package, sélectionnez Delete package (Supprimer le package).
5. Lorsque vous êtes invité à confirmer la suppression, sélectionnez Delete package (Supprimer le package).

Suppression d'une version de package (console)

Vous pouvez utiliser la console 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, sélectionnez Distributor.
3. Sur la page d'accueil de Distributor, sélectionnez le package dont vous souhaitez supprimer une version.
4. Sur la page des versions du package, sélectionnez la version à supprimer et sélectionnez Delete version (Supprimer la version).
5. Lorsque vous êtes invité à confirmer la suppression, sélectionnez 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 & macOS

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 figurer 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 figurer dans la liste.

```
aws ssm list-documents ^
    --filters Key=Name,Values=package-name
```

PowerShell

Pour supprimer un package (Tools for 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 figurer 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 & macOS

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 la 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 la 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 (Tools for 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 la 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 de la Référence de Command AWS CLI. 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

Vous pouvez utiliser AWS CloudTrail pour auditer l'activité liée à Distributor, une fonctionnalité de AWS Systems Manager. Pour de plus amples informations sur les options d'audit et de journalisation de Systems Manager, veuillez consulter Surveillance AWS Systems Manager.

Auditer l'activité de Distributor en utilisant CloudTrail

CloudTrail capture les appels d'API effectués dans la console AWS Systems Manager, l'AWS Command Line Interface (AWS CLI) et le kit SDK Systems Manager. Les informations peuvent être consultées dans la console CloudTrail ou stockées 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 Run Command et State Manager présentent les activités de création de documents, d'installation et de désinstallation de packages. Run Command et State Manager sont des fonctionnalités de AWS Systems Manager. 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' AWS Systems Manager API avec AWS CloudTrail.

Résolution des problèmes liés à AWS Systems ManagerDistributor

Les informations suivantes peuvent vous aider à résoudre les problèmes susceptibles de survenir lorsque vous utilisez Distributor une fonctionnalité de AWS Systems Manager.

Rubriques

Un package erroné portant le même nom est installé

Problème : vous avez installé un package, mais Distributor en a installé un autre à la place.

Cause : lors de l'installation, Systems Manager recherche les packages publiés par AWS avant les packages externes définis par l'utilisateur. Si le nom de package défini par l'utilisateur est identique au nom d'un package AWS publié, le AWS package est installé à la place de votre package.

Solution : Pour éviter ce problème, donnez à votre package un nom différent du nom d'un package AWS publié.

Erreur : impossible de récupérer le manifeste ; la dernière version du package est introuvable

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 de SSM Agent avec Distributor qui est antérieure à la version 2.3.274.0.

Solution : mettez à jour l'SSM Agent vers la version 2.3.274.0 ou ultérieure. Pour plus d'informations, consultez Mise à jour de SSM Agent à l'aide de Run Command ou Démonstration : Mise à jour automatique de l'SSM Agent (CLI).

Erreur : Impossible de récupérer le 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 de SSM Agent avec Distributor qui est antérieure à la version 2.3.274.0.

Solution : mettez à jour l'SSM Agent vers la version 2.3.274.0 ou ultérieure. Pour plus d'informations, consultez Mise à jour de SSM Agent à l'aide de Run Command ou Démonstration : Mise à jour automatique de l'SSM Agent (CLI).

Le package n'est pas pris en charge (l'action d'installation du package est manquante)

Problème : vous avez reçu une erreur similaire à la suivante.

```
Package is not supported (package is missing install action)
```

Cause : la structure du répertoire du package est incorrecte.

Solution : ne zippez pas un répertoire parent contenant le logiciel et les scripts requis. Au lieu de cela, créez un fichier `.zip` de tous les contenus requis directement dans le chemin absolu. Pour vérifier que la création du fichier `.zip` est correcte, dézippez le répertoire de la plateforme cible et vérifiez la structure du répertoire. Par exemple, le chemin absolu du script d'installation doit être `/`*`ExamplePackage_targetPlatform`*`/install.sh`.

Erreur : échec de téléchargement du manifeste : le document avec le nom n'existe pas

Problème : vous avez reçu une erreur similaire à la suivante.

```
Failed to download manifest - failed to retrieve package document description:
 InvalidDocument: Document with name filename does not exist.
```

Cause : Distributorne peut pas trouver le package par son nom lorsqu'un package Distributor est partagé à partir d'un autre compte.

Solution : lors du partage d'un package à partir d'un autre compte, utilisez l'Amazon Resource Name (ARN) complet pour le package et pas seulement son nom.

Chargement échoué.

Problème : vous avez reçu une erreur similaire à la suivante.

```
Upload failed. At least one of your files was not successfully uploaded to your S3
 bucket.
```

Cause : le nom de votre package de logiciels inclut une espace. Par exemple, le téléchargement de `Hello World.msi` échouerait.

AWS Systems Manager Ressources partagées

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

Rubriques

- AWS Systems Manager Documents

AWS Systems Manager Documents

Un document AWS Systems Manager (SSM) définit les actions exécutées par Systems Manager sur vos instances gérées. Systems Manager inclut plus d'une centaine de documents préconfigurés que vous pouvez utiliser en spécifiant des paramètres lors de l'exécution. Vous pouvez trouver des documents préconfigurés dans la console Documents de Systems Manager en sélectionnant l'onglet Propriété d'Amazon ou en spécifiant Amazon comme filtre `Owner` lors de l'appel de l'opération d'API `ListDocuments`. Les documents utilisent JSON (JavaScript Object Notation) ou YAML et incluent les étapes et paramètres que vous spécifiez. Pour vos premiers pas dans SSM documents, ouvrez Systems Manager console. Dans le panneau de navigation, cliquez sur Documents.

En quoi la fonction Documents peut-elle être utile à mon organisation ?

Documents, une fonction de AWS Systems Manager, offre les avantages suivants :

- Catégories de documents

  Pour vous aider à trouver les documents dont vous avez besoin, sélectionnez une catégorie en fonction du type de document que vous recherchez. Pour élargir votre recherche, vous pouvez choisir plusieurs catégories du même type de document. Le choix de catégories de différents types de documents n'est pas pris en charge. Les catégories ne sont prises en charge que pour les documents appartenant à Amazon.

- Version de document

  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 AWS Command Line Interface (AWS CLI) ou les appels d'API.

- Personnaliser les documents selon vos besoins

  Si vous souhaitez personnaliser les étapes et les actions dans un document, vous pouvez créer votre propre document. Le système stocke le document avec votre Compte AWS dans la Région AWS dans laquelle vous l'avez créé. Pour de plus amples informations sur la création d'un document SSM, veuillez consulter Création du contenu du document SSM.

- Baliser les documents

  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 politique AWS Identity and Access Management (IAM) qui spécifie les balises auxquelles un utilisateur ou un groupe peuvent accéder. Pour de plus amples informations, veuillez consulter Balisage des documents Systems Manager.

- Partager les documents

  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 Amazon Elastic Compute Cloud (Amazon 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 de plus amples informations, veuillez consulter Partage de documents SSM.

Qui devrait utiliser les documents ?

- Tout client AWS qui souhaite utiliser les fonctionnalités de Systems Manager pour améliorer son efficacité opérationnelle à grande échelle, réduire les erreurs associées aux interventions manuelles et accélérer la résolution des problèmes courants.
- Experts en infrastructure qui souhaitent automatiser les tâches de déploiement et de configuration.

- Les administrateurs qui souhaitent résoudre de manière fiable les problèmes courants, améliorer l'efficacité du dépannage et réduire les opérations répétitives.
- Les utilisateurs qui souhaitent automatiser une tâche qu'ils exécutent normalement manuellement.

Quels sont les types de documents SSM ?

Le tableau suivant décrit les différents types de documents SSM et leur utilisation.

Type	A utiliser avec	Détails
ApplicationConfiguration ApplicationConfigurationSchema	AWS AppConfig	AWS AppConfig, une fonctionnalité de AWS Systems Manager, permet de créer, gérer et déployer rapidement des configurations d'application. Vous pouvez stocker des données de configuration dans un document SSM en créant un document qui utilise le type de document `ApplicationConfiguration`. Pour plus d'informations, consultez Configurations libres dans le Guide de l'utilisateur AWS AppConfig. Si vous créez une configuration dans un document SSM, vous devez spécifier un schéma JSON correspondant. Le schéma utilise le type de document `ApplicationConfigurationSchema` et, comme un ensemble de règles, définit les propriétés autorisées pour chaque

Type	A utiliser avec	Détails
		paramètre de configuration de l'application. Pour de plus amples informations, veuillez consulter À propos des validateurs dans le Guide de l'utilisateur AWS AppConfig .
Runbook Automation	Automation State Manager Maintenance Windows	Utilisez des runbooks 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 runbooks Automation pour appliquer une configuration. Ces actions peuvent être exécutées sur une ou plusieurs cibles lors du cycle de vie d'une instance. Maintenance Windows utilise des runbooks Automation pour effectuer les tâches courantes de maintenance et de déploiement en fonction de la planification spécifiée. Tous les runbooks Automation pris en charge sur les systèmes d'exploitation Linux sont également pris en charge sur les instances EC2 pour macOS.

Type	A utiliser avec	Détails
Document Change Calendar	Change Calendar	Change Calendar, une fonctionnalité de AWS Systems Manager, utilise le type de document `ChangeCalendar` . Un document Change Calendar stocke une entrée de calendrier et les événements associés qui peuvent autoriser ou empêcher les actions Automation de modifier votre environnement. Dans Change Calendar, un document stocke les données iCalendar 2.0 au format texte brut. Change Calendar n'est pas pris en charge sur les instances EC2 pour macOS.

Type	A utiliser avec	Détails
Modèle AWS CloudFormation	AWS CloudFormation	Ces modèles AWS CloudFormation décrivent les ressources que vous voulez mettre en service dans vos piles CloudFormation. Le stockage de modèles CloudFormation sous forme de documents Systems Manager vous permet de bénéficier des fonctions des documents Systems Manager. Cela inclut la création et la comparaison de plusieurs versions de votre modèle, ainsi que le partage de votre modèle avec d'autres comptes dans la même Région AWS. Vous pouvez créer et modifier des modèles et des piles CloudFormation en utilisant Application Manager, une fonctionnalité de Systems Manager. Pour de plus amples informations, veuillez consulter Utilisation des modèles et des modèles AWS CloudFormation dans Application Manager..

Type	A utiliser avec	Détails
Document de commande	Run Command State Manager Maintenance Windows	Run Command, une fonctionnalité de AWS Systems Manager, utilise les documents Command pour exécuter des commandes. State Manager, une fonctionnalité de AWS Systems Manager utilise les documents Command pour appliquer une configuration. Ces actions peuvent être exécutées sur une ou plusieurs cibles lors du cycle de vie d'une instance. Maintenance Windows, une des fonctions de AWS Systems Manager, utilise des documents de commande pour appliquer une configuration en fonction de la planification spécifiée. La plupart des documents Command sont pris en charge sur tous les systèmes d'exploitation Linux et Windows Server pris en charge par Systems Manager. Les documents Command suivants sont pris en charge sur les instances EC2 pour macOS : • `AWS-ConfigureAWSPackage` • `AWS-RunPatchBaseline`

Type	A utiliser avec	Détails
		• `AWS-RunPatchBaselineAssociation` • `AWS-RunShellScript`
Modèle de pack de conformité AWS Config	AWS Config	Les modèles de packs de conformité AWS Config sont des documents au format YAML utilisés pour créer des packs de conformité contenant la liste des règles gérées ou personnalisées AWS Config et des actions de remédiation. Pour plus d'informations, consultez Pack de conformité.
Document de package	Distributor	Dans Distributor, une fonctionnalité de AWS Systems Manager, 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. Distributor n'est pas pris en charge sur les instances gérées Oracle Linux et macOS.

Type	A utiliser avec	Détails
Document de politique	State Manager	Inventory, une des fonctions de AWS Systems Manager, utilise le document de politique `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 runbooks Automation et les documents Command constituent la méthode privilégiée pour appliquer une politique à une instance gérée. Systems Manager Inventory et le document de politique `AWS-GatherSoftwareInventory` sont pris en charge sur tous les systèmes d'exploitation pris en charge par Systems Manager.

Type	A utiliser avec	Détails
Modèle d'analyse post-incident	Analyse post-incident d'Incident Manager	Incident Manager utilise le modèle d'analyse post-incident pour créer une analyse basée sur les bonnes pratiques de gestion des opérations AWS. Utilisez le modèle pour créer une analyse qui aidera votre équipe à identifier les améliorations à apporter à votre réponse aux incidents.

Type	A utiliser avec	Détails
Document de session	Session Manager	Session Manager, une fonctionnalité de AWS Systems Manager, utilise des documents 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. Les documents Session sont pris en charge sur tous les systèmes d'exploitation Linux et Windows Server pris en charge par Systems Manager. Les documents Command suivants sont pris en charge sur les instances EC2 pour macOS : • `AWS-PasswordReset` • `AWS-StartInteractiveCommand` • `AWS-StartPortForwardingSession` • `AWS-StartPortForwardingSessionToSocket` • `AWS-StartSSHSession`

Quotas de documents SSM

Pour plus d'informations sur les quotas de document SSM, veuillez consulter la rubrique Quotas de service Systems Manager dans la Référence générale d'Amazon Web Services.

Rubriques

Composants de document

Cette section comprend des informations sur les éléments qui composent les documents SSM.

Table des matières

Schémas, fonctionnalités et exemples

Les documents (SSM) AWS Systems Manager 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 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

Fonction	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 plugins dans le même document SSM. Le support multiplateforme utilise le même paramètre `precondition` dans une étape.

> **Note**
>
> Vous devez conserver SSM Agent AWS Systems Manager sur vos instances à jour avec la dernière version pour utiliser les nouvelles fonctions Systems Manager, ainsi que les fonctions de document SSM. Pour de plus amples informations, veuillez consulter Mise à jour de SSM Agent à l'aide de Run Command.

Le tableau suivant répertorie les différences entre les versions majeures du schéma.

Version 1.2	Version 2.2 (dernière version)	Détails
runtimeConfig	mainSteps	Dans la version 2.2, la section `mainSteps` remplace `runtimeConfig`. La section `mainSteps` permet à Systems Manager d'exécuter des étapes en séquence.
propriétés	inputs	Dans la version 2.2, la section `inputs` remplace la section `properties`. La section `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'une modification de nom.
ne s'applique pas	name	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 plugin ou pour valider les paramètres d'entrée que vous avez définis dans votre document SSM. Le paramètre `precondition` prend en charge le référencement des paramètres d'entrée de votre document SSM, et le `platformType` en utilisant les valeurs de`Linux`, `MacOS` et `Windows`. Seul l'opérateur `StringEquals` est pris en charge.

Pour les documents utilisant la version de schéma 2.2 ou une version ultérieure, si `precondition` n'est pas spécifié, chaque plugin est soit exécuté, soit ignoré en fonction de sa compatibilité avec le système d'exploitation. La compatibilité du plugin avec le système d'exploitation est évaluée avant la `precondition`. 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 plugin `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 plug-in `aws:runPowerShellScript` pour exécuter une commande PowerShell 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:
      timeoutSeconds: '60'
      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": {
            "timeoutSeconds": "60",
            "runCommand": [
               "Write-Output {{Message}}"
            ]
         }
      }
   ]
}
```

Exemples 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 plugins. Le support multiplateforme utilise le même paramètre `precondition` dans une étape, tel que décrit dans l'exemple suivant. Vous pouvez également utiliser le paramètre `precondition` pour valider les

paramètres d'entrée que vous avez définis dans votre document SSM. Cela apparaît dans le second des exemples suivants.

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"
            ]
         }
      }
   ]
}
```

YAML

```
---
schemaVersion: '2.2'
parameters:
  action:
    type: String
    allowedValues:
    - Install
    - Uninstall
  confirmed:
    type: String
    allowedValues:
    - True
    - False
mainSteps:
- action: aws:runShellScript
  name: InstallAwsCLI
```

```
  precondition:
    StringEquals:
    - "{{ action }}"
    - "Install"
  inputs:
    runCommand:
    - sudo apt install aws-cli
- action: aws:runShellScript
  name: UninstallAwsCLI
  precondition:
    StringEquals:
    - "{{ action }} {{ confirmed }}"
    - "Uninstall True"
  inputs:
    runCommand:
    - sudo apt remove aws-cli
```

JSON

```
{
   "schemaVersion": "2.2",
   "parameters": {
      "action": {
         "type": "String",
         "allowedValues": [
            "Install",
            "Uninstall"
         ]
      },
      "confirmed": {
         "type": "String",
         "allowedValues": [
            true,
            false
         ]
      }
   },
   "mainSteps": [
      {
         "action": "aws:runShellScript",
         "name": "InstallAwsCLI",
         "precondition": {
            "StringEquals": [
```

```
                "{{ action }}",
                "Install"
            ]
         },
         "inputs": {
            "runCommand": [
               "sudo apt install aws-cli"
            ]
         }
      },
      {
         "action": "aws:runShellScript",
         "name": "UninstallAwsCLI",
         "precondition": {
            "StringEquals": [
               "{{ action }} {{ confirmed }}",
               "Uninstall True"
            ]
         },
         "inputs": {
            "runCommand": [
               "sudo apt remove aws-cli"
            ]
         }
      }
   ]
}
```

Exemple de version de schéma 2.2 State Manager

Vous pouvez utiliser le document SSM suivant avec State Manager, une des fonctionnalités de Systems Manager, pour télécharger et installer le logiciel antivirus ClamAV. State Manager applique une configuration spécifique, ce qui signifie qu'à chaque fois que l'association State Manager est exécutée, le système vérifie si le logiciel ClamAV 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 de version de schéma 2.2 **`AWS-ConfigureAWSPackage`**

L'exemple suivant présente le document `AWS-ConfigureAWSPackage`. La section `mainSteps` inclut le plugin `aws:configurePackage` à l'étape `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."
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."
      }
   },
   "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 de version de schéma 1.2 **`aws:runShellScript`**

L'exemple suivant montre le document SSM `AWS-RunShellScript`. La section runtimeConfig inclut le plugin `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 runbook de version de schéma 0.3 ou ultérieur au format JSON.

```
{
    "description": "document-description",
    "schemaVersion": "0.3",
    "assumeRole": "{{assumeRole}}",
    "parameters": {
        "parameter1": {
            "type": "String",
            "description": "parameter-1-description",
            "default": ""
        },
        "parameter2": {
            "type": "String",
            "description": "parameter-2-description",
            "default": ""
        }
    },
    "variables": {
        "variable1": {
            "type": "StringMap",
            "description": "variable-1-description",
            "default": {}
        },
        "variable2": {
            "type": "String",
            "description": "variable-2-description",
            "default": "default-value"
        }
    },
    "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 runbook Automation YAML

L'exemple suivant montre le contenu d'un runbook 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 runbook 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: python3.8
      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: python3.8
    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
```

Éléments de données et paramètres

Cette rubrique décrit les éléments de données utilisés dans les documents SSM. La version du schéma utilisée pour créer un document définit la syntaxe et les éléments de données que le document accepte. Nous vous recommandons l'utilisation de la version de schéma 2.2 ou celle ultérieure pour les documents de Commande. Les runbooks Automation utilisent la version de schéma 0.3. De plus, les runbooks 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 relatives à l'utilisation de Markdown, consultez Utilisation de Markdown dans la console dans le Guide de mise en route AWS Management Console .

La section suivante décrit les éléments de données pouvant être inclut dans un document SSM.

Éléments de données niveau supérieur

schemaVersion

Version de schéma à utiliser.

Type : Version

Obligatoire : oui

description

Informations que vous fournissez pour décrire l'objectif du document. Vous pouvez également utiliser ce champ pour spécifier si un paramètre nécessite une valeur pour qu'un document s'exécute ou si la fourniture d'une valeur pour le paramètre est facultative. Les paramètres obligatoires et facultatifs peuvent être consultés dans les exemples de cette rubrique.

Type : chaîne

Obligatoire : non

parameters

Structure qui définit les paramètres acceptés par le document.

Pour les paramètres que vous utilisez souvent, nous vous recommandons de les stocker dans Parameter Store une fonctionnalité de AWS Systems Manager. Ensuite, définissez les paramètres de votre document faisant référence aux paramètres Parameter Store comme valeur par défaut. Pour référencer un paramètre Parameter Store, utilisez la syntaxe suivante.

```
{{ssm:parameter-name}}
```

Utilisez un paramètre faisant référence à un paramètre Parameter Store similaire à tout autre paramètre du document. Dans l'exemple suivant, la valeur par défaut du paramètre `commands` est le paramètre Parameter Store `myShellCommands`. En spécifiant le paramètre `commands` en tant que chaîne `runCommand`, le document exécute les commandes stockées dans le paramètre `myShellCommands`.

YAML

```
---
schemaVersion: '2.2'
description: runShellScript with command strings stored as Parameter Store
 parameter
parameters:
  commands:
    type: StringList
    description: "(Required) The commands to run on the instance."
    default: ["{{ ssm:myShellCommands }}"]
mainSteps:
- action: aws:runShellScript
  name: runShellScriptDefaultParams
  inputs:
    runCommand:
    - "{{ commands }}"
```

JSON

```
{
    "schemaVersion": "2.2",
    "description": "runShellScript with command strings stored as Parameter Store
 parameter",
    "parameters": {
      "commands": {
        "type": "StringList",
        "description": "(Required) The commands to run on the instance.",
        "default": ["{{ ssm:myShellCommands }}"]
```

```
      }
    },
    "mainSteps": [
      {
        "action": "aws:runShellScript",
        "name": "runShellScriptDefaultParams",
        "inputs": {
            "runCommand": [
              "{{ commands }}"
          ]
        }
      }
    ]
  }
```

> **Note**
>
> Vous pouvez référencer les paramètres `String` et `StringList` Parameter Store dans la section `parameters` d'un document. Vous ne pouvez pas référencer les paramètres Parameter Store `SecureString`.

Pour plus d'informations sur Parameter Store, consultez AWS Systems Manager Parameter Store.

Type : Structure

La structure `parameters` accepte les champs et valeurs suivants :

- `type` : (Obligatoire) les valeurs autorisées sont : `String`, `StringList`, `Integer`, `Boolean`, `MapList` et `StringMap`. Pour consulter des exemples de chaque type, consultez Exemples de paramètres `type` de document SSM dans la section suivante.

  > **Note**
  >
  > Les documents de type Command ne prennent en charge que les types de paramètres `String` et `StringList`.

- `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.

  > **Note**
  >
  > Systems Manager effectue deux validations pour `allowedPattern`. La première validation est effectuée à l'aide de la bibliothèque Java regex au niveau de l'API lorsque vous utilisez un document. La deuxième validation est effectuée sur SSM Agent en utilisant la bibliothèque Go regexpavant de traiter le document.

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 le AWS Management Console. `textfield`est une zone de texte d'une seule ligne. `textarea`est une zone de texte multiligne.
- `minItems` : (Facultatif) Nombre minimum d'éléments autorisés.
- `maxItems` : (Facultatif) Nombre maximum d'éléments autorisés.
- `minChars` : (Facultatif) Nombre minimum d'éléments autorisés.
- `maxChars` : (Facultatif) Nombre maximum de caractères de paramètre autorisés.

Obligatoire : non

variables

(Schéma version 0.3 uniquement) Valeurs que vous pouvez référencer ou mettre à jour tout au long des étapes d'un runbook d'Automation. Les variables sont similaires aux paramètres, mais diffèrent d'une manière très importante. Les valeurs des paramètres sont statiques dans le contexte d'un runbook, mais les valeurs des variables peuvent être modifiées dans le contexte du runbook. Lors de la mise à jour de la valeur d'une variable, le type de données doit correspondre au type de données défini. Pour plus d'informations sur la mise à jour des valeurs de variables dans une automatisation, veuillez consulter [`aws:updateVariable` : met à jour la valeur d'une variable runbook].

Type : booléen | Entier | | Chaîne MapList | | StringList StringMap

Obligatoire : non

YAML

```
variables:
    payload:
        type: StringMap
        default: "{}"
```

JSON

```
{
    "variables": [
        "payload": {
            "type": "StringMap",
            "default": "{}"
        }
    ]
}
```

runtimeConfig

(Version de schéma 1.2 seulement) Configuration de l'instance telle qu'appliquée par un ou plusieurs plugins Systems Manager. L'exécution en séquence des plugins 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 en crée un nouveauAMI,

vous pouvez spécifier « »CreateImage. ImageId« comme valeur de sortie, puis utilisez cette sortie pour créer de nouvelles instances lors d'une exécution d'automatisation ultérieure. Pour plus d'informations sur les sorties, consultez Utilisation des sorties d'action comme entrées.

Type : Dictionnaire<String, > OutputConfiguration

Obligatoire : non

files

(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 runbooks Automation prennent en charge les scripts pour Python 3.7, Python 3.8, PowerShell Core 6.0 et PowerShell 7.0. Pour de plus amples informations sur l'inclusion de scripts dans les runbooks Automation, veuillez consulter Utilisation de scripts dans des runbooks et Créer des runbooks à l'aide de Document Builder.

Lorsque vous créez un runbook d'automatisation avec des pièces jointes, vous devez également spécifier les fichiers joints à l'aide de l'`--attachments`option (pour AWS CLI) ou `Attachments` (pour l'API et le SDK). Vous pouvez spécifier l'emplacement du fichier tant pour les fichiers locaux que pour les fichiers stockés dans des compartiments Amazon Simple Storage Service (Amazon S3). Pour plus d'informations, consultez la section Pièces jointes dans le Guide de référence de AWS Systems Manager l'API.

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 paramètres **`type`** 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. Cela est également vrai lorsque vous utilisez des paramètres en aval dans les entrées pour d'autres plugins ou actions AWS Systems Manager d'automatisation. Par exemple, vous ne pouvez pas référencer un paramètre `StringList` dans l'entrée `documentParameters` du plugin `aws:runDocument`. L'entrée `documentParameters` accepte une carte de chaînes même si le type de paramètre de document SSM en aval est un paramètre `StringList` et correspond au paramètre auquel vous faites référence.

Lorsque vous utilisez des paramètres avec des actions 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.

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: "(Optional) The target EC2 instance ID."
```

JSON

```
"InstanceId":{
  "type":"String",
  "description":"(Optional) 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."
  default: ""
  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 », ni 0.

YAML

```
---
canRun:
  type: Boolean
  description: ''
  default: true
```

JSON

```
"canRun": {
  "type": "Boolean",
  "description": "",
  "default": true
}
```

Entier

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. Les clés et les valeurs doivent être des chaînes. Par exemple, {"Env": "Prod"}.

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' StringMap objets.

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
}
```

Affichage du contenu du document SSM Command

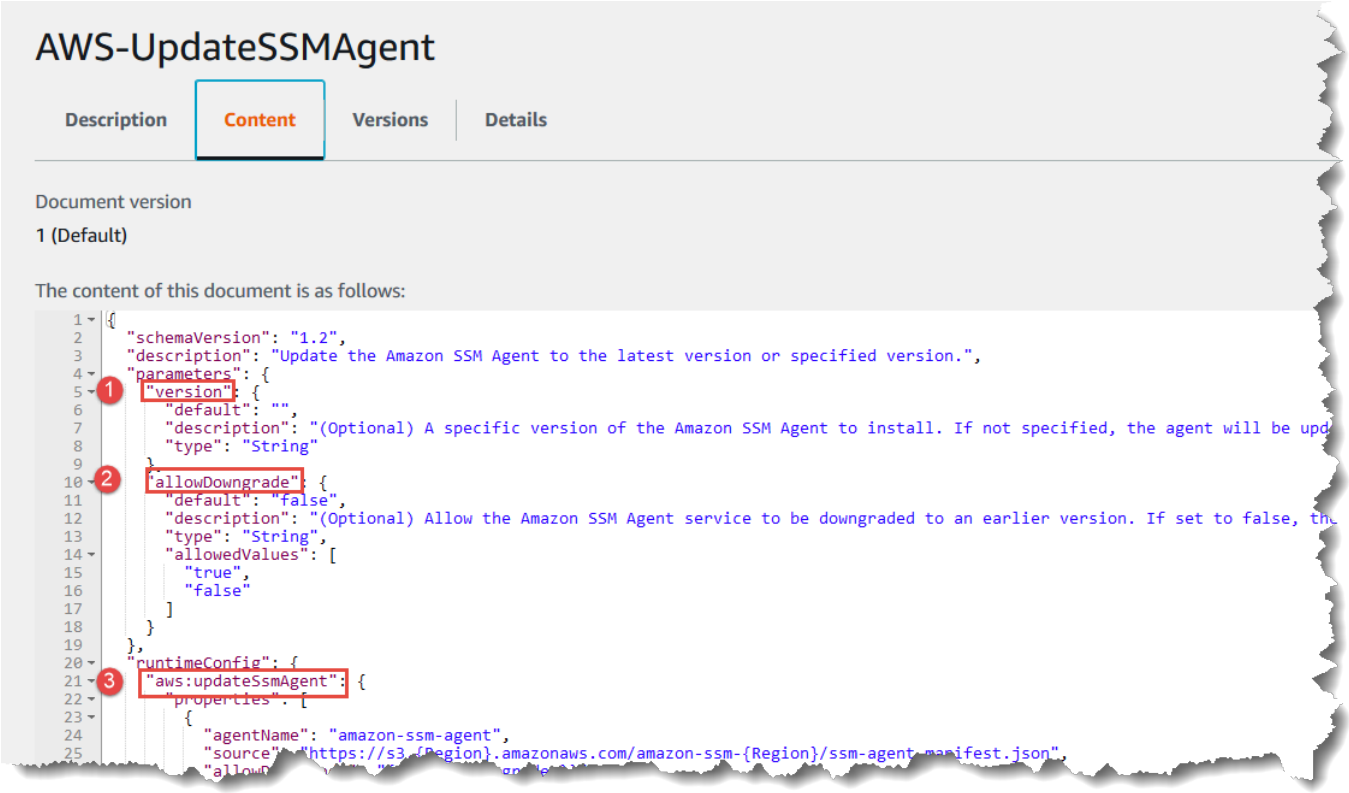
Pour prévisualiser les paramètres obligatoires et facultatifs d'un document de commande AWS Systems Manager (SSM), outre les actions exécutées par le document, vous pouvez consulter le contenu du document dans la console Systems Manager.

Pour afficher le contenu d'un document SSM Command

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.
3. Dans la zone de recherche, sélectionnez Type de document, puis sélectionnez Command.
4. Sélectionnez le nom d'un document, puis l'onglet Content (Contenu).

5. Dans le champ Contenu, vérifiez les paramètres disponibles et les étapes d'action du document.

   Par exemple, l'image suivante montre que (1) `version` et (2) `allowDowngrade` sont des paramètres facultatifs pour le document `AWS-UpdateSSMAgent`, et que la première action exécutée par le document est (3) `aws:updateSsmAgent`.



Référence de plug-in de document Command

Cette référence décrit les plugins que vous pouvez spécifier dans un document de type commande AWS Systems Manager (SSM). Ces plugins ne peuvent pas être utilisés dans les runbooks Automation SSM qui utilisent des actions Automation. Pour plus d'informations sur les actions AWS Systems Manager d'automatisation, consultezRéférence sur les actions Systems Manager Automation.

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

plugins Systems Manager. Pour plus d'informations sur les documents, la création de documents et les différences entre les versions de schéma, consultez AWS Systems Manager Documents.

> **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. Les plugins de document suivants sont pris en charge sur les instances Amazon Elastic Compute Cloud (Amazon EC2) pour macOS :
>
> - `aws:refreshAssociation`
> - `aws:runShellScript`
> - `aws:runPowerShellScript`
> - `aws:softwareInventory`
> - `aws:updateSsmAgent`

Table des matières

- aws:updateSsmAgent

Entrées partagées

Avec SSM Agent version 3.0.502 et ultérieure uniquement, tous les plugins peuvent utiliser les entrées suivantes :

finallyStep

La dernière étape que le document doit exécuter. Si cette entrée est définie pour une étape, elle a priorité sur une valeur `exit` spécifiée dans les entrées `onFailure` ou `onSuccess`. Pour qu'une étape avec cette entrée s'exécute comme prévu, elle doit être la dernière étape définie dans les `mainSteps` de votre document.

Type : booléen

Valeurs valides : `true` | `false`

Obligatoire : non

onFailure

Si vous spécifiez cette entrée pour un plugin avec la valeur `exit` et que l'étape échoue, le statut de l'étape reflète l'échec et le document n'exécute pas les étapes restantes sauf si une `finallyStep` a été définie. Si vous spécifiez cette entrée pour un plugin avec la valeur `successAndExit` et que l'étape échoue, le statut de l'étape affiche la réussite et le document n'exécute pas les étapes restantes sauf si une `finallyStep` a été définie.

Type : chaîne

Valeurs valides : `exit` | `successAndExit`

Obligatoire : non

onSuccess

Si vous spécifiez cette entrée pour un plugin et que l'étape s'exécute correctement, le document n'exécute pas les étapes restantes sauf si une `finallyStep` a été définie.

Type : chaîne

Valeurs valides : `exit`

Obligatoire : non

YAML

```
---
schemaVersion: '2.2'
description: Shared inputs example
parameters:
  customDocumentParameter:
    type: String
    description: Example parameter for a custom Command-type document.
mainSteps:
- action: aws:runDocument
  name: runCustomConfiguration
  inputs:
    documentType: SSMDocument
    documentPath: "yourCustomDocument"
    documentParameters: '"documentParameter":{{customDocumentParameter}}'
    onSuccess: exit
- action: aws:runDocument
  name: ifConfigurationFailure
  inputs:
    documentType: SSMDocument
    documentPath: "yourCustomRepairDocument"
    onFailure: exit
- action: aws:runDocument
  name: finalConfiguration
  inputs:
    documentType: SSMDocument
    documentPath: "yourCustomFinalDocument"
    finallyStep: true
```

JSON

```
{
   "schemaVersion": "2.2",
   "description": "Shared inputs example",
   "parameters": {
      "customDocumentParameter": {
         "type": "String",
         "description": "Example parameter for a custom Command-type document."
      }
   },
   "mainSteps":[
      {
```

```
         "action": "aws:runDocument",
         "name": "runCustomConfiguration",
         "inputs": {
            "documentType": "SSMDocument",
            "documentPath": "yourCustomDocument",
            "documentParameters": "\"documentParameter\":
{{customDocumentParameter}}",
            "onSuccess": "exit"
         }
      },
      {
         "action": "aws:runDocument",
         "name": "ifConfigurationFailure",
         "inputs": {
            "documentType": "SSMDocument",
            "documentPath": "yourCustomRepairDocument",
            "onFailure": "exit"
         }
      },
      {
         "action": "aws:runDocument",
         "name":"finalConfiguration",
         "inputs": {
            "documentType": "SSMDocument",
            "documentPath": "yourCustomFinalDocument",
            "finallyStep": true
         }
      }
   ]
}
```

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.

Syntaxe

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 }}"
            }
         ]
      }
   }
}
```

Propriétés

action

Action à effectuer.

Type : énumération

Valeurs valides : `Install` | `Repair` | `Uninstall`

Obligatoire : oui

parameters

Paramètres pour le programme d'installation.

Type : chaîne

Obligatoire : non

source

URL du fichier `.msi` pour l'application.

Type : chaîne

Obligatoire : oui

sourceHash

Hachage SHA256 du fichier `.msi`.

Type : chaîne

Obligatoire : non

`aws:cloudWatch`

Exportez des données depuis Windows Server Amazon CloudWatch ou Amazon CloudWatch Logs et surveillez-les à l'aide de CloudWatch métriques. Ce plug-in s'exécute uniquement sur les systèmes d'exploitation Windows Server. Pour plus d'informations sur la configuration de CloudWatch l'intégration avec Amazon Elastic Compute Cloud (Amazon EC2), consultez la section Collecter des métriques, des journaux et des traces avec CloudWatch l'agent dans le guide de l'utilisateur Amazon CloudWatch .

> **Important**
>
> L' CloudWatch agent unifié a été remplacé SSM Agent en tant qu'outil d'envoi des données de journal à Amazon CloudWatch Logs. Le plugin SSM Agent aws:cloudWatch n'est pas pris en charge. Nous vous recommandons de n'utiliser que l' CloudWatch agent unifié pour vos processus de collecte de journaux. Pour plus d'informations, consultez les rubriques suivantes :
>
> - Envoi des journaux des nœuds vers CloudWatch des journaux unifiés (CloudWatch agent)
> - Migrer la collecte des journaux des nœuds Windows Server vers l' CloudWatch agent
> - Collecte de métriques, de journaux et de traces avec l' CloudWatch agent dans le guide de CloudWatch l'utilisateur Amazon.

Vous pouvez exporter et contrôler les types de données suivants :

ApplicationEventJournal

Envoie les données du journal des événements de l'application à CloudWatch Logs.

CustomLogs

Envoie n'importe quel fichier journal sous forme de texte à Amazon CloudWatch Logs. Le CloudWatch plugin crée une empreinte digitale 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 plugin 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 les données de suivi des événements pour Windows (ETW) aux CloudWatch journaux.

IIS

Envoie les données du journal IIS à CloudWatch Logs.

PerformanceCounter

Envoie les compteurs de performance Windows à. CloudWatch Vous pouvez sélectionner différentes catégories dans lesquelles vous souhaitez effectuer le téléchargement CloudWatch sous forme de statistiques. Pour chaque compteur de performance que vous souhaitez télécharger, créez une PerformanceCountersection avec un identifiant unique (par exemple, « PerformanceCounter 2 », « PerformanceCounter 3 », etc.) et configurez ses propriétés.

> **Note**
>
> Si le plugin AWS Systems Manager SSM Agent ou le CloudWatch plugin est arrêté, les données du compteur de performance ne sont pas enregistrées CloudWatch. Ce

comportement est différent des journaux personnalisés ou des journaux d'événements Windows. Les journaux personnalisés et les journaux d'événements Windows préservent les données des compteurs de performance et les CloudWatch téléchargent SSM Agent une fois que le CloudWatch plug-in est disponible.

SecurityEventJournal

Envoie les données du journal des événements de sécurité à CloudWatch Logs.

SystemEventJournal

Envoie les données du journal des événements du 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 des identifiants uniques (par exempleCloudWatch, « 2 », CloudWatch 3, etc.) et spécifier une région différente pour chaque nouvel identifiant afin d'envoyer les mêmes données à différents emplacements.

CloudWatchJournaux

La destination où vos données de journaux sont envoyées. Vous pouvez ajouter d'autres sections avec des identifiants uniques (par exemple, « CloudWatch Logs2 CloudWatchLogs », 3, etc.) et spécifier une région différente pour chaque nouvel identifiant afin d'envoyer les mêmes données à différents emplacements.

Syntaxe

```
"runtimeConfig":{
        "aws:cloudWatch":{
            "settings":{
                "startType":"{{ status }}"
            },
            "properties":"{{ properties }}"
        }
    }
```

Paramètres et propriétés

AccessKey

Votre ID de clé d'accès . 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 : chaîne

Obligatoire : non

CategoryName

Catégorie du compteur de performances en provenance de Performance Monitor.

Type : chaîne

Obligatoire : oui

CounterName

Nom du compteur de performances en provenance de Performance Monitor.

Type : chaîne

Obligatoire : oui

CultureName

Paramètres régionaux où l'horodatage est consigné. S'il CultureNameest vide, il utilise par défaut les mêmes paramètres régionaux que ceux utilisés par votre Windows Server instance.

Type : chaîne

Valeurs valides : pour obtenir une liste des valeurs prises en charge, consultez Support des langues nationales sur le site Web de Microsoft. Les valeurs div, div-MV, hu, et hu-HU ne sont pas prises en charge.

Obligatoire : non

DimensionName

Une dimension pour votre CloudWatch métrique Amazon. Si vous spécifiez `DimensionName`, vous devez spécifier `DimensionValue`. Ces paramètres offrent une autre vue lors de la création de listes de métriques. Vous pouvez utiliser la même dimension pour plusieurs métriques afin d'afficher toutes les métriques appartenant à une dimension spécifique.

Type : chaîne

Obligatoire : non

DimensionValue

Une valeur de dimension pour votre CloudWatch métrique Amazon.

Type : chaîne

Obligatoire : non

Encodage

Encodage de fichier à utiliser (par exemple, UTF-8). Utilisez le nom d'encodage, pas le nom complet.

Type : chaîne

Valeurs valides : pour obtenir une liste des valeurs prises en charge, veuillez consulter la rubrique Classe d'encodage dans la bibliothèque Microsoft Learn (langue française non garantie).

Obligatoire : oui

Filtre

Préfixe des noms de journaux. Laissez ce paramètre vide de façon à surveiller tous les fichiers.

Type : 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, ainsi que la destination des données (CloudWatch ou CloudWatch journaux). Par exemple, pour envoyer un compteur de performance défini sous `"Id": "PerformanceCounter"` vers la CloudWatch destination définie sous`"Id": "CloudWatch"`, entrez «PerformanceCounter, CloudWatch ». De même, pour envoyer le journal personnalisé, le journal ETW et le journal système vers la destination CloudWatch des journaux définie ci-dessous`"Id": "ETW"`, entrez « (ETW), CloudWatch Logs ». En outre, vous pouvez envoyer le même compteur de performances ou fichier journal à plus d'une destination. Par

exemple, pour envoyer le journal de l'application vers deux destinations différentes que vous avez définies `"Id": "CloudWatchLogs"` ci-dessous`"Id": "CloudWatchLogs2"`, saisissez « ApplicationEvent Log, (CloudWatchLogs, CloudWatchLogs 2) ».

Type : chaîne

Valeurs valides (source) : `ApplicationEventLog` | `CustomLogs` | `ETW` | `PerformanceCounter` | `SystemEventLog` | `SecurityEventLog`

Valeurs valides (destination) : `CloudWatch` | `CloudWatchLogs` | `CloudWatch`*n* | `CloudWatchLogs`*n*

Obligatoire : oui

FullName

Nom complet du composant.

Type : chaîne

Obligatoire : oui

Id

Identifie la source ou la destination des données. Cet identificateur doit être unique au sein du fichier de configuration.

Type : 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 : chaîne

Obligatoire : oui

Niveaux

Les types de messages à envoyer à Amazon CloudWatch.

Type : chaîne

Valeurs valides :

- 1 - Uniquement les messages d'erreur chargés.
- 2 - Uniquement les messages d'avertissement chargés.
- 4 - Uniquement les messages d'information chargés.

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 : entier

Obligatoire : non

LogDirectoryParcours

Pour CustomLogs, le chemin 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 : chaîne

Obligatoire : oui

LogGroup

Nom de votre groupe de journaux. Ce nom est affiché sur l'écran Log Groups de la CloudWatch console.

Type : 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, sélectionnez Applications and Services Logs (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 voulez télécharger (par exemple, Microsoft > Windows > Sauvegarde > Opérationnelle), puis cliquez sur Create Custom View (Créer une vue personnalisée).
3. Dans la boîte de dialogue Create Custom View (Créer une vue personnalisée), sélectionnez l'onglet XML. Cela LogNamese trouve dans la balise <Select Path=> (par exemple,`Microsoft-Windows-Backup`). Copiez ce texte dans le LogNameparamètre.

Type : chaîne

Valeurs valides : `Application` | `Security` | `System` | `Microsoft-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 : chaîne

Valeurs valides : `{instance_id}` | `{hostname}` | `{ip_address}` *`<log_stream_name>`*

Si vous entrez un nom de flux de journal qui n'existe pas encore, 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 du flux de journal spécifié dans ce paramètre est affiché sur l'écran Log Groups > Streams for *< YourLog Stream>* de la CloudWatch console.

Obligatoire : oui

MetricName

CloudWatch Mesure dans laquelle vous souhaitez que les données de performance soient incluses.

> **Note**
>
> N'utilisez pas de caractères spéciaux dans ce nom. Sinon, la métrique et les alarmes associées risquent de ne pas fonctionner.

Type : chaîne

Obligatoire : oui

NameSpace

Namespace de la métrique où vous voulez écrire les données des compteurs de performances.

Type : chaîne

Obligatoire : oui

PollInterval

Nombre de secondes avant qu'un nouveau compteur de performances et des données de journal soient chargés.

Type : entier

Valeurs valides : indiquez une valeur égale ou supérieure à 5 secondes. La valeur recommandée est 15 secondes (00:00:15).

Obligatoire : oui

Région

L' Région AWS endroit où vous souhaitez envoyer les données du journal. 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 : chaîne

Valeurs valides : les identifiants de région Régions AWS pris en charge à la fois par Systems Manager et CloudWatch Logs`us-east-2`, tels que`eu-west-1`, et`ap-southeast-1`. Pour obtenir la liste des points de terminaison Régions AWS pris en charge par chaque service, consultez Amazon CloudWatch Logs Service Endpoints et Systems Manager dans le. 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 : chaîne

Obligatoire : non

startType

Activez ou CloudWatch désactivez l'instance.

Type : 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 : chaîne

Obligatoire : oui

TimeZoneGentil

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 aucune information de

fuseau horaire, CloudWatch Logs utilise par défaut le fuseau horaire local. Ce paramètre est ignoré si votre horodatage contient déjà des informations de fuseau horaire.

Type : chaîne

Valeurs valides : `Local` | `UTC`

Obligatoire : non

Unité

Unité de mesure appropriée pour la métrique.

Type : chaîne

Valeurs valides : Secondes | Microsecondes | Millisecondes | Octets | Kilo-octets | Mégaoctets | Gigaoctets | Téraoctets | Bits | Kilobits | Mégabits | Gigabits | Térabits | Pourcentage | Nombre | Octets/seconde | Kilo-octets/seconde | Mégaoctets/seconde | Gigaoctets/seconde | Téraoctets/seconde | Bits/seconde | Kilobits/seconde | Mégabits/seconde | Gigabits/seconde | Térabits/seconde | Nombre/seconde | Aucune.

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.

Syntaxe

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 : énumération

Valeurs valides : `Install` | `Uninstall`

Obligatoire : oui

`aws:configurePackage`

(Schema version 2.0 ou ultérieure) Installez ou désinstallez un AWS Systems Manager Distributor package. 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 AWS packages disponibles pour Windows Server incluent les suivants : `AWSPVDriver` `AWSNVMeAwsEnaNetworkDriver`,`AwsVssComponents`,`AmazonCloudWatchAgent`,`CodeDeployAgent`, et `AWSSupport-EC2Rescue.`

AWS Les packages disponibles pour les systèmes d'exploitation Linux sont les suivants : `AmazonCloudWatchAgentCodeDeployAgent`, et`AWSSupport-EC2Rescue`.

Syntaxe

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
  ssmParameter:
```

```
    description: "(Required) Argument stored in Parameter Store."
    type: String
    default: "{{ ssm:parameter_store_arg }}"
mainSteps:
- action: aws:configurePackage
  name: configurePackage
  inputs:
    name: "{{ name }}"
    action: "{{ action }}"
    additionalArguments:
      "{\"SSM_parameter_store_arg\": \"{{ ssmParameter }}\", \"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"
         ]
      },
      "ssmParameter": {
         "description": "(Required) Argument stored in Parameter Store.",
         "type": "String",
         "default": "{{ ssm:parameter_store_arg }}"
      }
   },
   "mainSteps": [
      {
         "action": "aws:configurePackage",
         "name": "configurePackage",
```

```
         "inputs": {
            "name": "{{ name }}",
            "action": "{{ action }}",
            "additionalArguments": "{\"SSM_parameter_store_arg\":
 \"{{ ssmParameter }}\", \"SSM_custom_arg\": \"myValue\"}"
         }
      }
   ]
}
```

Inputs

name

Nom du AWS package à installer ou à désinstaller. Les packages disponibles sont les suivants : `AWSPVDriver`, `AwsEnaNetworkDriver`, `AwsVssComponents` et `AmazonCloudWatchAgent`.

Type : chaîne

Obligatoire : oui

action

Installe ou désinstalle un package.

Type : énumération

Valeurs valides : `Install` | `Uninstall`

Obligatoire : oui

Type d'installation

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 AWS publiés. `Uninstall and reinstall`est la valeur par défaut.

Type : énumération

Valeurs valides : `Uninstall and reinstall` | `In-place update`

Obligatoire : non

AdditionalArguments

Une chaîne JSON de paramètres supplémentaires à ajouter à vos scripts d'installation, de désinstallation ou de mise à jour. Chaque paramètre doit être préfixé avec `SSM_`. Vous pouvez référencer un paramètre Parameter Store dans vos arguments supplémentaires en utilisant la convention `{{ssm:`*`parameter-name`*`}}`. Pour utiliser le paramètre supplémentaire dans vos scripts d'installation, de désinstallation ou de mise à jour, vous devez référencer le paramètre en tant que variable d'environnement en utilisant la syntaxe qui correspond au système d'exploitation. Par exemple, dans PowerShell, vous référencez l'`SSM_arg`argument comme`$Env:SSM_arg`. Le nombre d'arguments que vous définissez n'est pas limité, mais la saisie d'un argument supplémentaire est limitée à 4 096 caractères. Cette limite inclut l'ensemble des clés et des 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 : 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. Ce plugin modifie le nom d'hôte des instances Linux au format EC2AMAZ-*XXXXXXX*. Pour plus d'informations sur la jonction d'instances EC2, voir Joindre une instance EC2 à votre répertoire AWS Microsoft AD géré dans le Guide d'AWS Directory Service administration.

Syntaxe

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
  directoryOU:
    description: "(Optional) The organizational unit to assign the computer object
 to."
    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"
    },
    "directoryOU": {
        "description": "(Optional) The organizational unit to assign the computer
 object to.",
        "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 }}"
         }
      }
   }
}
```

Propriétés

directoryId

ID du répertoire.

Type : chaîne

Obligatoire : oui

Exemple : "directoryId": "d-1234567890"

directoryName

Nom du domaine.

Type : chaîne

Obligatoire : oui

Exemple : "directoryName": "example.com"

directoryOU

Unité d'organisation (UO).

Type : chaîne

Obligatoire : non

Exemple : "directoryOU": "OU=test,DC=example,DC=com"

dns IpAddresses

Adresses IP des serveurs DNS.

Type : StringList

Obligatoire : oui

Exemple : « dns « : [" 198.51.100.1", IpAddresses "198.51.100.2"]

Exemples

Pour obtenir des exemples, consultez la rubrique Join an Amazon EC2 Instance to your (Joindre une instance Amazon EC2 à votre) AWS Managed Microsoft AD dans le Guide d'administration AWS Directory Service .

`aws:downloadContent`

(Schema version 2.0 ou ultérieure) Téléchargez des documents et des scripts SSM depuis des sites distants. GitHub Enterpriseles référentiels ne sont pas pris en charge. Ce plug-in est pris en charge sur les systèmes d'exploitation Linux et Windows Server.

Syntaxe

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 du téléchargement. Systems Manager prend en charge les types de source suivants pour le téléchargement de scripts et de documents SSM : `GitHub`, `Git`, `HTTP`, `S3` et `SSMDocument`.

Type : 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 :

- owner: propriétaire du référentiel.
- repository: nom du référentiel.
- path: chemin d'accès au fichier ou au répertoire que vous souhaitez télécharger.
- getOptions : options supplémentaires pour récupérer le contenu d'une branche autre que master ou d'un commit spécifique dans le référentiel. getOptions peut être omise si vous utilisez la dernière validation dans la branche maître. Si votre référentiel a été créé après le 1er octobre 2020, la branche par défaut peut être nommée « main » au lieu de « master ». Dans ce cas, vous devrez spécifier des valeurs pour le paramètre getOptions.

  Ce paramètre utilise le format suivant :

  - branch:refs/heads/*`branch_name`*

    L'argument par défaut est `master`.

    Pour spécifier une branche autre que celle par défaut, utilisez le format suivant :

    branch:refs/heads/*`branch_name`*
  - commitID:*`commitID`*

    L'argument 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. Par exemple :

    ```
    "getOptions": "commitID:bbc1ddb94...b76d3bEXAMPLE",
    ```
- TokenInfo : le paramètre Systems Manager (SecureStringun paramètre) dans lequel vous stockez les informations de GitHub votre jeton d'accès, au format. `{{ssm-secure:`*`secure-string-token-name`*`}}`

> **Note**
>
> Ce `tokenInfo` champ est le seul champ du plugin de document SSM qui prend en charge un SecureString paramètre. SecureString les paramètres ne sont pris en charge pour aucun autre champ, ni pour aucun autre plugin de document SSM.

```
{
    "owner":"TestUser",
    "repository":"GitHubTest",
    "path":"scripts/python/test-script",
    "getOptions":"branch:master",
    "tokenInfo":"{{ssm-secure:secure-string-token}}"
}
```

Pour sourceType **`Git`**, vous devez spécifier les informations suivantes :

- référentiels

  L'URL du référentiel Git vers le fichier ou le répertoire que vous souhaitez télécharger.

  Type : chaîne

En outre, vous pouvez préciser les paramètres facultatifs suivants :

- getOptions

  Options supplémentaires pour récupérer le contenu d'une branche autre que master ou d'un commit spécifique dans le référentiel. getOptions peut être omise si vous utilisez la dernière validation dans la branche maître.

  Type : chaîne

  Ce paramètre utilise le format suivant :

  - branch:refs/heads/*branch_name*

    L'argument par défaut est `master`.

    `"branch"` n'est requis que si votre document SSM est stocké dans une branche autre que `master`. Par exemple :

```
"getOptions": "branch:refs/head/main"
```

- commitID:*commitID*

  L'argument 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. Par exemple :

```
"getOptions": "commitID:bbc1ddb94...b76d3bEXAMPLE",
```

- privateSSHKey

  Clé SSH à utiliser lors de la connexion au `repository` spécifié. Vous pouvez utiliser le format suivant pour référencer un paramètre `SecureString` pour la valeur de votre clé SSH : `{{ssm-secure:`*your-secure-string-parameter*`}}`.

  Type : chaîne

- ignorer la HostKey vérification

  Détermine la valeur de l' StrictHostKeyCheckingoption lors de la connexion à celle `repository` que vous spécifiez. La valeur par défaut est `false`.

  Type : booléen

- nom d'utilisateur

  Nom d'utilisateur à utiliser lors de la connexion au `repository` spécifié avec HTTP. Vous pouvez utiliser le format suivant pour référencer un paramètre `SecureString` pour la valeur de votre nom d'utilisateur : `{{ssm-secure:`*your-secure-string-parameter*`}}`.

  Type : chaîne

- mot de passe

  Mot de passe à utiliser lors de la connexion au `repository` spécifié avec HTTP. Vous pouvez utiliser le format suivant pour référencer un paramètre `SecureString` pour la valeur de votre mot de passe : `{{ssm-secure:`*your-secure-string-parameter*`}}`.

  Type : chaîne

Pour sourceType **`HTTP`**, vous devez spécifier les informations suivantes :

- url

  L'URL du fichier ou du répertoire que vous souhaitez télécharger.

  Type : chaîne

En outre, vous pouvez préciser les paramètres facultatifs suivants :

- autoriser InsecureDownload

  Détermine si un téléchargement peut être effectué sur une connexion non chiffrée avec Secure Socket Layer (SSL) ou Transport Layer Security (TLS). La valeur par défaut est `false`. Nous vous déconseillons d'effectuer des téléchargements sans chiffrement. Si vous choisissez de le faire, vous en assumez les risques associés. La sécurité est une responsabilité partagée entre vous AWS et vous. Cela est décrit comme un modèle de responsabilité partagée. Pour en savoir plus, veuillez consulter le Modèle de responsabilité partagée.

  Type : booléen

- authMethod

  Détermine si un nom d'utilisateur et un mot de passe sont utilisés pour l'authentification lors de la connexion au `url` spécifié. Si vous spécifiez `Basic` ou `Digest`, vous devez fournir des valeurs pour les paramètres `username` et `password`. Pour utiliser la méthode `Digest`, SSM Agent version 3.0.1181.0 ou une version ultérieure doit être installé sur votre instance. La méthode `Digest` prend en charge le chiffrement MD5 et SHA256.

  Type : chaîne

  Valeurs valides : `None` | `Basic` | `Digest`

- nom d'utilisateur

  Nom d'utilisateur à utiliser lors de la connexion au `url` spécifié avec l'authentification `Basic`. Vous pouvez utiliser le format suivant pour référencer un paramètre `SecureString` pour la valeur de votre nom d'utilisateur : `{{ssm-secure:`*`your-secure-string-parameter`*`}}`.

  Type : chaîne

- mot de passe

Mot de passe à utiliser lors de la connexion au `url` spécifié avec l'authentification `Basic`. Vous pouvez utiliser le format suivant pour référencer un paramètre `SecureString` pour la valeur de votre mot de passe : `{{ssm-secure:`*`your-secure-string-parameter`*`}}`.

Type : chaîne

Pour sourceType **`S3`**, spécifiez les informations suivantes :

- path: URL du fichier ou du répertoire que vous voulez télécharger à partir d'Amazon S3.

```
{
    "path": "https://s3.amazonaws.com/DOC-EXAMPLE-BUCKET/powershell/
helloPowershell.ps1"
}
```

Pour sourceType **`SSMDocument`**, spécifiez l'une des options suivantes :

- name: nom et version du document au format suivant : `name:version`. La version est facultative.

```
{
    "name": "Example-RunPowerShellScript:3"
}
```

- name : 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 : chaîne

Obligatoire : non

aws:psModule

Installez PowerShell des modules sur une instance Amazon EC2. Ce plug-in s'exécute uniquement sur les systèmes d'exploitation Windows Server.

Syntaxe

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 }}"
            }
         ]
      }
   }
}
```

Propriétés

runCommand

PowerShell Commande à exécuter après l'installation du module.

Type : StringList

Obligatoire : non

source

URL ou chemin d'accès sur l'instance du fichier `.zip` de l'application.

Type : chaîne

Obligatoire : oui

sourceHash

Hachage SHA256 du fichier `.zip`.

Type : chaîne

Obligatoire : non

timeoutSeconds

Nombre de secondes accordées à l'exécution d'une commande avant qu'elle soit considérée comme ayant échoué.

Type : chaîne

Obligatoire : non

workingDirectory

Chemin d'accès au répertoire de travail sur votre instance.

Type : 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.

Syntaxe

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 ID d'association. 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.

Syntaxe

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 : chaîne

Obligatoire : oui

conteneur

ID du conteneur Docker.

Type : chaîne

Obligatoire : non

image

Nom de l'image Docker.

Type : chaîne

Obligatoire : non

cmd

Commande du conteneur.

Type : chaîne

Obligatoire : non

memory

Limite de mémoire du conteneur.

Type : chaîne

Obligatoire : non

cpuShares

Parts d'UC du conteneur (poids relatif).

Type : chaîne

Obligatoire : non

volume

Montages de volume du conteneur.

Type : StringList

Obligatoire : non

env

Variables d'environnement du conteneur.

Type : chaîne

Obligatoire : non

utilisateur

Nom d'utilisateur du conteneur.

Type : chaîne

Obligatoire : non

publish

Ports publiés du conteneur.

Type : 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 plugin avec le plugin `aws:downloadContent` 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. Ce plugin ne prend pas en charge l'exécution du document `AWS-UpdateSSMAgent` ou de tout autre document utilisant le plugin `aws:updateSsmAgent`.

Syntaxe

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 : 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 : chaîne

Obligatoire : non

documentParameters

Paramètres pour le document.

Type : StringMap

Obligatoire : non

`aws:runPowerShellScript`

Exécutez PowerShell des scripts ou spécifiez le chemin d'accès au script à exécuter. Ce plug-in s'exécute sur les systèmes d'exploitation Linux et Microsoft Windows Server.

Syntaxe

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 }}"
            }
         ]
      }
   }
}
```

Propriétés

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 : chaîne

Obligatoire : non

workingDirectory

Chemin d'accès au répertoire de travail sur votre instance.

Type : 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.

Syntaxe

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 }}"
            }
         ]
      }
   }
}
```

Propriétés

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 : chaîne

Obligatoire : non

workingDirectory

Chemin d'accès au répertoire de travail sur votre instance.

Type : 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 AWS Systems Manager State Manager association. Systems Manager collecte les données d'inventaire lorsque l'association est exécutée. Si vous ne créez pas l'association en premier et essayez d'appeler le plugin `aws:softwareInventory`, le système renvoie l'erreur suivante :

```
The aws:softwareInventory plugin can only be invoked via ssm-associate.
```

Une instance ne peut avoir qu'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 AWS Systems Manager Inventory.

Syntaxe

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 : chaîne

Obligatoire : non

awsComponents

(Facultatif) Collectez des métadonnées pour AWS des composants tels que amazon-ssm-agent.

Type : chaîne

Obligatoire : non

files

(Facultatif, nécessite la version de l'SSM Agent 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.

Type : chaîne

Obligatoire : non

networkConfig

(Facultatif) Collectez les métadonnées pour les configurations de réseau.

Type : chaîne

Obligatoire : non

windowsUpdates

(Facultatif) Collectez les métadonnées pour toutes les mises à jour Windows.

Type : chaîne

Obligatoire : non

instance DetailedInformation

(Facultatif) Collectez plus d'informations d'instance que ce qui est fourni par le plugin 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 : chaîne

Obligatoire : non

services

(Facultatif, système d'exploitation, Windows nécessite uniquement la version de l'SSM Agent 2.2.64.0 ou une version ultérieure) Collectez les métadonnées pour les configurations de service.

Type : chaîne

Obligatoire : non

windowsRegistry

(Facultatif, système d'exploitation, Windows nécessite uniquement la version de l'SSM Agent 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.

Type : chaîne

Obligatoire : non

windowsRoles

(Facultatif, système d'exploitation, Windows nécessite uniquement la version de l'SSM Agent 2.2.64.0 ou une version ultérieure) Collectez les métadonnées pour les configurations de rôle Microsoft Windows.

Type : 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.

Type : chaîne

Obligatoire : non

`aws:updateAgent`

Mise à jour du service EC2Config avec la version la plus récente ou spécification d'une version plus ancienne. Ce plug-in s'exécute uniquement sur les systèmes d'exploitation Microsoft Windows Server. Pour plus d'informations sur le service EC2Config, consultez Configuration d'une instance Windows à l'aide du service EC2Config (ancien) dans le guide de l'utilisateur Amazon EC2.

Syntaxe

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 }}"
         }
      }
   }
}
```

Propriétés

agentName

EC2Config. Nom de l'agent qui exécute le service EC2Config.

Type : chaîne

Obligatoire : oui

allowDowngrade

Permet au service EC2Config de revenir à 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 : chaîne

Obligatoire : oui

targetVersion

Version spécifique du service EC2Config à installer. Si vous ne spécifiez pas de version, le service est mis à jour avec la dernière version.

Type : chaîne

Obligatoire : non

`aws:updateSsmAgent`

Mise à jour de l'SSM Agent à la version la plus récente ou spécification d'une version plus ancienne. Ce plug-in s'exécute sur les systèmes d'exploitation Linux et Windows Server. Pour plus d'informations, consultez Utilisation de l'option SSM Agent.

Syntaxe

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": "awsupdateSsmAgent",
      "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 }}"
            }
         ]
      }
   }
}
```

Propriétés

agentName

amazon-ssm-agent. Nom de l'agent Systems Manager qui traite les demandes et exécute les commandes sur l'instance.

Type : chaîne

Obligatoire : oui

allowDowngrade

Permet à l'SSM Agent 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'SSM Agent à installer. Vous ne pouvez pas modifier cet emplacement.

Type : chaîne

Obligatoire : oui

targetVersion

Version spécifique de SSM Agent à installer. Si vous ne spécifiez pas de version, l'agent est mis à jour avec la dernière version.

Type : chaîne

Obligatoire : non

Création du contenu du document SSM

Si les documents AWS Systems Manager publics n'exécutent pas toutes les actions que vous souhaitez effectuer sur vos AWS ressources, vous pouvez créer vos propres documents SSM. Vous pouvez également cloner des documents SSM en utilisant la console. Le clonage de documents

copie le contenu d'un document existant vers un nouveau document que vous pouvez modifier. Lors de la création ou du clonage d'un document, le contenu du document ne doit pas dépasser 64 Ko. Ce quota inclut également le contenu spécifié pour les paramètres d'entrée lors de l'exécution. 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, plugins et syntaxe disponibles pour les documents SSM. Nous vous recommandons de vous familiariser avec les ressources suivantes.

- Rédaction de vos propres AWS Systems Manager documents
- Éléments de données et paramètres
- Schémas, fonctionnalités et exemples
- Référence de plug-in de document Command
- Référence sur les actions Systems Manager Automation
- Variables système Automation
- Exemples supplémentaires de runbook
- Utilisation de runbooks Systems Manager Automation avec la AWS Toolkit for Visual Studio Code
- Créer des runbooks à l'aide de Document Builder
- Utilisation de scripts dans des runbooks

AWS les documents SSM prédéfinis peuvent effectuer certaines des actions dont vous avez besoin. Vous pouvez appeler ces documents en utilisant les plugins `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 & macOS

```
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
```

Clonage d'un document SSM

Vous pouvez cloner AWS Systems Manager des documents à l'aide de la console Systems Manager Documents pour créer des documents SSM. Le clonage de documents SSM copie le contenu d'un document existant vers un nouveau document que vous pouvez modifier. Vous ne pouvez pas cloner un document dont la taille est supérieure à 64 Ko.

Pour cloner un document SSM

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.
3. Dans la zone de recherche, saisissez le nom du document que vous voulez cloner.
4. Sélectionnez le nom du document à cloner, puis sélectionnez Clone document (Cloner un document) dans le menu déroulant Actions.
5. Modifiez le document selon vos préférences, puis sélectionnez Create document (Créer un document) pour enregistrer le document.

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

- Création de documents composites

Création de documents composites

Un document composite AWS Systems Manager (SSM) est un document personnalisé qui exécute 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 Comptes AWS au même endroit Région AWS pour réduire la maintenance des documents SSM et garantir la cohérence.

Par exemple, vous pouvez créer un document composite qui effectue les actions suivantes :

1. Installe tous les correctifs de la liste autorisée.
2. Installer un logiciel antivirus.
3. Télécharge des scripts depuis GitHub et les exécute.

Dans cet exemple, votre document SSM personnalisé inclut les plugins suivants pour effectuer ces actions :

1. Le plugin `aws:runDocument` pour exécuter le document `AWS-RunPatchBaseline`, qui installe tous les correctifs autorisés répertoriés.
2. Le plugin `aws:runDocument` pour exécuter le document `AWS-InstallApplication`, qui installe le logiciel antivirus.
3. Le `aws:downloadContent` plugin depuis lequel 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. 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.

Pour créer un document composite, ajoutez le plugin `aws:runDocument` 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` plugin pour télécharger un document SSM depuis un dépôt GitHub public vers un répertoire local appelé bootstrap. Le document SSM s'appelle StateManagerBootstrap .yml (un document YAML).

2. Exécute le `aws:runDocument` plugin pour exécuter le document StateManagerBootstrap .yml. Aucun paramètre n'est spécifié.
3. Exécute le plugin `aws:runDocument` pour exécuter le document SSM `AWS-ConfigureDocker pre-defined`. 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\"}"
      }
    }
  ]
```

```
}
```

Plus d'informations

- Pour plus d'informations sur le redémarrage des serveurs et des instances en appelant des scripts à l'aide de Run Command, consultez Gestion des redémarrages lors de l'exécution de commandes.
- Pour plus d'informations sur les plugins que vous pouvez ajouter à un document SSM personnalisé, consultez Référence de plug-in de document Command.
- Si vous souhaitez simplement exécuter un document à partir d'un emplacement distant (sans créer de document composite), consultez Exécution de documents à partir d'emplacements distants.

Utilisation de documents

Cette section inclut des informations sur l'utilisation des documents SSM.

Table des matières

Utilisation de documents SSM dans des associations State Manager

Si vous créez un document SSM pour State Manager une fonctionnalité de AWS Systems Manager, vous devez associer le document à vos instances gérées après l'avoir ajouté au système. Pour plus d'informations, consultez Utilisation d'associations dans Systems Manager.

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 plugins contradictoires (par exemple, joindre au domaine et supprimer du domaine), c'est le dernier plugin exécuté qui correspondra à l'état final. State Manager ne valide pas la séquence logique ou la cohérence des commandes ou des plugins 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 document SSM 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 document SSM avec vous ultérieurement, vos associations remapper automatiquement à celui-ci.

Comparaison des versions de documents SSM

Vous pouvez comparer les différences de contenu entre les versions de documents AWS Systems Manager (SSM) dans la console Systems Manager Documents. Lorsque vous comparez des versions d'un document SSM, les différences entre le contenu des versions sont mises en surbrillance.

Pour comparer le contenu d'un document SSM (console)

1. Ouvrez la AWS Systems Manager console 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, chosissez le document dont vous voulez comparer le contenu.
4. Sous l'onglet Content (Contenu), sélectionnez Compare versions (Comparer des versions), puis sélectionnez la version du document auquel vous voulez comparer le contenu.

Créer un document SSM (console)

Après avoir créé le contenu pour votre document SSM personnalisé, comme décrit dans Rédaction du contenu du document SSM, 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 AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.
3. Sélectionnez Create command or session (Créer une commande ou une session).
4. Saisissez 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 plus d'informations sur le balisage des ressources Systems Manager, consultez Balisage des ressources Systems Manager.
9. Sélectionnez Create document pour enregistrer le document.

Créer un document SSM (ligne de commande)

Après avoir créé le contenu de votre document personnalisé AWS Systems Manager (SSM), comme décrit dansRédaction du contenu du document SSM, vous pouvez utiliser le AWS Command Line Interface (AWS CLI) ou créer un document SSM AWS Tools for PowerShell à partir de votre contenu. En voici un exemple dans la commande suivante :

Avant de commencer

Installez et configurez le AWS CLI ou le AWS Tools for PowerShell, si ce n'est pas déjà fait. Pour plus d'informations, consultez la section Installation ou mise à jour de la version la plus récente de l' AWS CLI et Installation d' AWS Tools for PowerShell.

Exécutez la commande suivante. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

Linux & macOS

```
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"
```

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)

Après avoir créé le contenu de votre document personnalisé AWS Systems Manager (SSM), comme décrit dansRédaction du contenu du document SSM, vous pouvez utiliser votre SDK préféré pour appeler l'opération d' AWS Systems Manager CreateDocumentAPI afin de créer un document SSM à partir de votre contenu. La chaîne JSON ou YAML du paramètre de demande `Content` est généralement lue à partir d'un fichier. Les exemples de fonction suivants créent un document SSM à l'aide des kits SDK 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 documents personnalisés, consultez Éléments de données et paramètres.

Suppression de documents SSM personnalisés

Si vous ne souhaitez plus utiliser de document SSM personnalisé, vous pouvez le supprimer à l'aide du AWS Command Line Interface (AWS CLI) ou de la AWS Systems Manager console.

Pour supprimer un document SSM (AWS CLI)

1. Avant de supprimer le document, nous vous recommandons de dissocier toutes les instances qui lui sont associées.

   Exécutez la commande suivante pour dissocier une instance d'un document.

   ```
   aws ssm delete-association --instance-id "123456789012" --name "documentName"
   ```

   Il n'y a pas de sortie si la commande réussit.

2. Exécutez la commande suivante. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

Linux

```
aws ssm delete-document \
    --name "document-name" \
    --document-version "document-version" \
    --version-name "version-name"
```

Windows

```
aws ssm delete-document ^
    --name "document-name" ^
    --document-version "document-version" ^
    --version-name "version-name"
```

PowerShell

```
Delete-SSMDocument `
    -Name "document-name" `
    -DocumentVersion 'document-version' `
    -VersionName 'version-name'
```

Il n'y a pas de sortie si la commande réussit.

> **Important**
>
> Si la `document-version` ou le `version-name` ne sont pas fournis, toutes les versions du document sont supprimées.

Pour supprimer un document SSM (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.
3. Sélectionnez le document que vous souhaitez supprimer.
4. Sélectionnez Delete (Supprimer). Lorsque vous êtes invité à supprimer le document, sélectionnez Delete (Supprimer).

Exécution de documents à partir d'emplacements distants

Vous pouvez exécuter des documents AWS Systems Manager (SSM) depuis des sites distants en utilisant le document SSM `AWS-RunDocument` prédéfini. Ce document prend en charge l'exécution de documents SSM stockés aux emplacements suivants :

- GitHubRéférentiels publics et privés (GitHub Enterprisenon pris en charge)
- Compartiments Amazon S3
- Systems Manager

Bien que vous puissiez également exécuter des documents distants en utilisant State Manager ou Automation, les fonctionnalités de AWS Systems Manager, la procédure suivante décrit uniquement comment exécuter des documents SSM distants AWS Systems Manager Run Command à l'aide de la console Systems Manager.

> **Note**
>
> `AWS-RunDocument` peut servir à exécuter des documents SSM de type commande, mais pas d'autres types du genre runbooks Automation. `AWS-RunDocument` utilise `aws:downloadContent`. Pour plus d'informations sur le plugin `aws:downloadContent`, consultez `aws:downloadContent`.

Avant de commencer

Avant d'exécuter un document distant, vous devez effectuer les tâches suivantes :

- Créez un document Command SSM et enregistrez-le dans un emplacement distant. Pour de plus amples informations, consultez Création du contenu du document SSM.
- Si vous envisagez d'exécuter un document distant stocké dans un GitHub référentiel privé, vous devez créer un `SecureString` paramètre Systems Manager pour votre jeton d'accès GitHub de sécurité. Vous ne pouvez pas accéder à un document distant dans un GitHub dépôt 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.

Exécuter un document distant (console)

Pour exécuter un document distant

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Run Command.
3. Sélectionnez Run Command (Exécuter la commande).
4. Dans la liste Document, sélectionnez **`AWS-RunDocument`**.
5. Dans Paramètres de commande, pour Type de source, sélectionnez une option.
   - Si vous le souhaitez GitHub, spécifiez les informations sur la source au format suivant :

```
{
    "owner": "owner_name",
    "repository": "repository_name",
    "path": "path_to_document",
    "getOptions":"branch:branch_name",
    "tokenInfo": "{{ssm-secure:secure-string-token}}"
}
```

Par exemple :

```
{
    "owner":"TestUser",
    "repository":"GitHubTestExamples",
    "path":"scripts/python/test-script",
    "getOptions":"branch:exampleBranch",
    "tokenInfo":"{{ssm-secure:my-secure-string-token}}"
}
```

> **Note**
>
> `getOptions` sont des options supplémentaires pour récupérer le contenu d'une branche autre que master ou d'un commit spécifique dans le référentiel. `getOptions` peut être omise si vous utilisez la dernière validation dans la branche maître. Le paramètre `branch` n'est requis que si votre document SSM est stocké dans une branche autre que `master`.

Pour utiliser la version d'un document SSM dans un commit particulier de votre référentiel, utilisez `commitID` avec `getOptions` au lieu de `branch`. Par exemple :

```
"getOptions": "commitID:bbc1ddb94...b76d3bEXAMPLE",
```

- Si vous sélectionnez S3, spécifiez les informations source au format suivant :

```
{"path":"URL_to_document_in_S3"}
```

Par exemple :

```
{"path":"https://s3.amazonaws.com/DOC-EXAMPLE-BUCKET/scripts/ruby/mySSMdoc.json"}
```

- Si vous sélectionnez SSMDocument, spécifiez les informations source au format suivant :

```
{"name": "document_name"}
```

Par exemple :

```
{"name": "mySSMdoc"}
```

6. Dans le champ Document Parameters (Paramètres du document), saisissez les paramètres du document SSM distant. Par exemple, si vous exécutez le document `AWS-RunPowerShell`, vous pouvez spécifier ce qui suit :

```
{"commands": ["date", "echo \"Hello World\""]}
```

Si vous exécutez le document `AWS-ConfigureAWSPack`, vous pouvez spécifier ce qui suit :

```
{
   "action":"Install",
   "name":"AWSPVDriver"
}
```

7. Dans la section Targets (Cibles), sélectionnez les nœuds gérés sur lesquels vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant des instances ou des appareils de périphérie manuellement ou en spécifiant un groupe de ressources.

> **Tip**
>
> Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés 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. Pour Rate control (Contrôle de débit) :

   - Dans Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage de nœuds gérés sur lesquels exécuter simultanément la commande.

     > **Note**
     >
     > Si vous avez sélectionné des cibles en spécifiant les balises appliquées aux nœuds gérés ou en spécifiant des groupes de ressources AWS , et que vous n'êtes pas certain du nombre de nœuds gérés ciblés, limitez le nombre de cibles autorisées à exécuter simultanément le document en indiquant un pourcentage.

   - Dans Error threshold (Seuil d'erreur), indiquez quand arrêter l'exécution de la commande sur les autres nœuds gérés après l'échec de celle-ci sur un certain nombre ou un certain pourcentage de nœuds. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les nœuds gérés sur lesquels la commande est toujours 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). Saisissez 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 (pour les instances EC2) ou de la

fonction du service IAM (pour les machines activées par un système hybride) attribués à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, assurez-vous que le profil d'instance ou le rôle de service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

11. Dans la section SNS notifications (Notifications SNS), si vous souhaitez envoyer des notifications sur le statut d'exécution des commandes, cochez la case Enable SNS notifications (Activer les notifications SNS).

    Pour plus d'informations sur la configuration des notifications Amazon SNS pour Run Command, consultez Surveillance des changements d'état du Systems Manager à l'aide des notifications Amazon SNS.

12. Cliquez sur Exécuter.

> **Note**
>
> Pour plus d'informations sur le redémarrage des serveurs et des instances en appelant des scripts à l'aide de Run Command, consultez Gestion des redémarrages lors de l'exécution de commandes.

Partage de documents SSM

Vous pouvez partager AWS Systems Manager (SSM) des documents en privé ou en public avec les comptes correspondants. 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 Compte AWS . Pour partager un document SSM publiquement, modifiez les autorisations du document et spécifiez `All`. Les documents ne peuvent pas être partagés simultanément publiquement et en privé.

> **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.

Limites

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 Modification des autorisations d'un document SSM partagé.
- Vous pouvez partager un document avec un maximum de 1 000 Comptes AWS. Vous pouvez demander que cette limite soit augmentée dans le Centre AWS Support. Pour Limit type (Type de limite), sélectionnez 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), sélectionnez EC2 Systems Manager et décrivez le motif de votre demande.
- Les documents ne peuvent être partagés avec d'autres comptes Région AWS que dans le même compte. Le partage inter-régions n'est pas pris en charge.

Pour plus d'informations sur les quotas de service Systems Manager, consultez Service Quotas AWS Systems Manager.

Table des matières

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

Vérifiez attentivement votre document AWS Systems Manager (SSM) et supprimez toutes les informations sensibles. Par exemple, vérifiez que le document ne contient pas vos AWS informations d'identification. 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.

Bloquer le partage public de documents

À moins que votre cas d'utilisation exige que le partage public soit autorisé, nous vous recommandons d'activer le paramètre de blocage du partage public pour vos documents SSM dans la section Preferences (Préférences) de la console Systems Manager Documents.

Limiter les actions Run Command à l'aide d'une politique d'approbation IAM

Créez une politique restrictive AWS Identity and Access Management (IAM) pour les utilisateurs qui auront accès au document. La politique IAM détermine les documents SSM qu'un utilisateur peut voir dans la console Amazon Elastic Compute Cloud (Amazon EC2) ou en `ListDocuments` appelant à l'aide du () ou. AWS Command Line Interface AWS CLI AWS Tools for Windows PowerShell La politique limite également les actions que l'utilisateur peut effectuer avec un document SSM. Vous pouvez créer une politique restrictive pour qu'un utilisateur ne puisse utiliser que des documents spécifiques. Pour plus d'informations, consultez Exemples de politiques gérées par le client.

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 approprié
- 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`. Le hachage ne peut pas vérifier le contenu du document à partir d'emplacements externes.

Bloquer le partage public de documents SSM

À moins que votre cas d'utilisation ne nécessite l'activation du partage public, nous vous recommandons d'activer le paramètre de blocage du partage public pour vos documents AWS Systems Manager (SSM). L'activation de ce paramètre empêche l'accès non souhaité à vos documents SSM. Le paramètre de blocage du partage public est un paramètre au niveau du compte qui peut varier d'un compte à l'autre Région AWS. Exécutez les tâches suivantes pour bloquer le partage public de vos documents SSM.

Bloquer le partage public (console)

Pour bloquer le partage public de vos documents SSM

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.
3. Sélectionnez Preferences (Préférences), puis Edit (Modifier) dans la section Block public sharing (Bloquer le partage public).
4. Cochez la case Block public sharing (Bloquer le partage public), puis sélectionnez Save (Enregistrer).

Bloquer le partage public (ligne de commande)

Ouvrez le AWS Command Line Interface (AWS CLI) ou AWS Tools for Windows PowerShell sur votre ordinateur local et exécutez la commande suivante pour bloquer le partage public de vos documents SSM.

Linux & macOS

```
aws ssm update-service-setting  \
    --setting-id /ssm/documents/console/public-sharing-permission \
    --setting-value Disable \
    --region 'The Région AWS you want to block public sharing in'
```

Windows

```
aws ssm update-service-setting ^
    --setting-id /ssm/documents/console/public-sharing-permission ^
    --setting-value Disable ^
    --region "The Région AWS you want to block public sharing in"
```

PowerShell

```
Update-SSMServiceSetting `
    -SettingId /ssm/documents/console/public-sharing-permission `
    -SettingValue Disable `
    -Region The Région AWS you want to block public sharing in
```

Confirmez que la valeur du paramètre a été mise à jour en utilisant la commande suivante.

Linux & macOS

```
aws ssm get-service-setting   \
    --setting-id /ssm/documents/console/public-sharing-permission \
    --region The Région AWS you blocked public sharing in
```

Windows

```
aws ssm get-service-setting  ^
    --setting-id /ssm/documents/console/public-sharing-permission ^
    --region "The Région AWS you blocked public sharing in"
```

PowerShell

```
Get-SSMServiceSetting `
    -SettingId /ssm/documents/console/public-sharing-permission `
    -Region The Région AWS you blocked public sharing in
```

Restreindre l'accès pour bloquer le partage public avec IAM

Vous pouvez créer des politiques AWS Identity and Access Management (IAM) qui empêchent les utilisateurs de modifier le paramètre de blocage du partage public. Cela empêche les utilisateurs d'autoriser l'accès non souhaité à vos documents SSM.

Voici un exemple de politique IAM qui empêche les utilisateurs de mettre à jour le paramètre de blocage du partage public. Pour utiliser cet exemple, vous devez remplacer l'exemple d'ID de compte Amazon Web Services par votre propre ID de compte.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": "ssm:UpdateServiceSetting",
            "Resource": "arn:aws:ssm:*:987654321098:servicesetting/ssm/documents/
console/public-sharing-permission"
        }
    ]
}
```

Partager un document SSM

Vous pouvez partager des documents AWS Systems Manager (SSM) à l'aide de la console Systems Manager. Lors du partage de documents depuis la console, seule la version par défaut du document peut être partagée. Vous pouvez également partager des documents SSM par programmation en appelant l'opération d'`ModifyDocumentPermission`API à l'aide du AWS Command Line Interface (AWS CLI) ou du AWS Tools for Windows PowerShell SDK. AWS Avant de partager un document, obtenez les ID de Compte AWS des personnes avec lesquelles vous voulez le partager. Vous devez spécifier ces ID de compte lorsque vous partagez le document.

Partager un document (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.
3. Dans la liste des documents, sélectionnez le document que vous souhaitez partager, puis sélectionnez Afficher les détails. Sur l'onglet Permissions (Autorisations), vérifiez que vous êtes le propriétaire du document. Seul le propriétaire d'un document peut le partager.
4. Sélectionnez Edit (Modifier).
5. Pour partager la commande publiquement, sélectionnez Public, puis Save. Pour partager la commande de façon privée, sélectionnez Private (Privé), saisissez l'ID de compte Compte AWS , puis sélectionnez Add permission (Ajouter une autorisation et Save (Enregistrer).

Partager un document (ligne de commande)

La procédure suivante nécessite que vous spécifiiez un Région AWS pour votre session de ligne de commande.

1. Ouvrez le AWS CLI ou AWS Tools for Windows PowerShell sur votre ordinateur local et exécutez la commande suivante pour spécifier vos informations d'identification.

   Dans la commande ci-après, remplacez *region* (région) par vos propres informations. Pour obtenir une liste des valeurs *region* prises en charge, consultez la colonne Région dans la rubrique Points de terminaison de service Systems Manager de la Référence générale d'Amazon Web Services.

   Linux & macOS

   ```
   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
   ```

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 & macOS

```
aws ssm list-documents
```

Windows

```
aws ssm list-documents
```

PowerShell

```
Get-SSMDocumentList
```

3. Utilisez la commande suivante pour obtenir un document spécifique.

Linux & macOS

```
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 & macOS

```
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 & macOS

   ```
   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. Vous pouvez éventuellement spécifier la version du document que vous souhaitez partager à l'aide du paramètre `--shared-document-version`. Si vous ne spécifiez pas de version, le système partage la version `Default` du document. Cet exemple de commande partage le document de façon privée avec la personne spécifiée, en fonction de son ID d' Compte AWS .

   Linux & macOS

   ```
   aws ssm modify-document-permission \
       --name document name \
       --permission-type Share \
       --account-ids-to-add Compte AWS ID
   ```

Windows

```
aws ssm modify-document-permission ^
    --name document name ^
    --permission-type Share ^
    --account-ids-to-add Compte AWS ID
```

PowerShell

```
Edit-SSMDocumentPermission `
    –Name document name `
    -PermissionType Share `
    -AccountIdsToAdd Compte AWS ID
```

7. Utilisez la commande suivante pour partager un document publiquement.

Linux & macOS

```
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 AWS Systems Manager (SSM) ou que vous 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.

Arrêt du partage d'un document (console)

Arrêt du partage d'un document

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.
3. Dans la liste des documents, choisissez le document que vous souhaitez arrêter de partager, puis sélectionnez Détails. Dans la section Autorisations, vérifiez que vous êtes le propriétaire du document. Seul le propriétaire d'un document peut arrêter le partage de celui-ci.
4. Sélectionnez Edit (Modifier).
5. Choisissez X pour supprimer l' Compte AWS identifiant qui ne devrait plus avoir accès à la commande, puis cliquez sur Enregistrer.

Arrêter le partage d'un document (ligne de commande)

Ouvrez le AWS CLI ou AWS Tools for Windows PowerShell sur votre ordinateur local et exécutez la commande suivante pour arrêter de partager une commande.

Linux & macOS

```
aws ssm modify-document-permission \
    --name document name \
    --permission-type Share \
    --account-ids-to-remove 'Compte AWS ID'
```

Windows

```
aws ssm modify-document-permission ^
    --name document name ^
    --permission-type Share ^
```

```
    --account-ids-to-remove "Compte AWS ID"
```

PowerShell

```
Edit-SSMDocumentPermission `
    -Name document name `
    -PermissionType Share `
    -AccountIdsToRemove Compte AWS ID
```

Utilisation de documents SSM partagés

Lorsque vous partagez un document AWS Systems Manager (SSM), le système génère un Amazon Resource Name (ARN) et l'attribue à la commande. Si vous sélectionnez et exécutez un document partagé à partir de la console Systems Manager, vous ne voyez pas l'ARN. Toutefois, si vous voulez exécuter un document SSM partagé à l'aide d'une méthode autre que la console Systems Manager, vous devez spécifier l'ARN complet du document pour le `DocumentName` paramètre de la requête. 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 des ARN pour les documents AWS publics (documents commençant par`AWS-*`) ou les documents dont vous êtes le propriétaire.

Utiliser un document SSM partagé (ligne de commande)

Pour répertorier tous les documents SSM publics

Linux & macOS

```
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 documents SSM privés qui ont été partagés avec vous

Linux & macOS

```
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 documents SSM disponibles

Linux & macOS

```
aws ssm list-documents
```

Windows

```
aws ssm list-documents
```

PowerShell

```
Get-SSMDocumentList
```

Pour obtenir des informations sur un document SSM qui a été partagé avec vous

Linux & macOS

```
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 document SSM partagé

Linux & macOS

```
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 documents SSM

Vous pouvez rechercher des documents SSM dans la banque de documents AWS Systems Manager (SSM) en utilisant une recherche en texte libre ou une recherche basée sur des filtres. Vous pouvez également ajouter des documents à vos favoris pour retrouver les documents SSM fréquemment utilisés. Les sections suivantes décrivent comment utiliser ces fonctions.

Utilisation de la recherche en texte libre

La zone de recherche de la page Documents de Systems Manager prend en charge la recherche en texte libre. La recherche en texte libre compare le ou les termes de recherche saisis au nom de document dans chaque document SSM. Si vous saisissez un seul terme de recherche, **`ansible`** par exemple, Systems Manager renvoie tous les documents SSM dans lesquels ce terme a été découvert. Si vous saisissez plusieurs termes de recherche, Systems Manager effectue une recherche en utilisant une instruction `OR`. Par exemple, si vous spécifiez **`ansible`** et **`linux`**, la recherche renvoie tous les documents dont le nom contient le mot-clé either (l'un ou l'autre).

Si vous saisissez un terme de recherche en texte libre et que vous sélectionnez une option de recherche telle que Platform type (Type de plateforme), la recherche utilise une instruction `AND` et renvoie tous les documents avec le mot clé dans leur nom et le type de plateforme spécifié.

> **Note**
>
> Notez les détails suivants sur la recherche en texte libre.
>
> - La recherche de texte libre recherche n'est pas sensible à la casse.
> - Les termes de recherche exigent 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 saisissez un espace entre les termes de recherche, le système inclut l'espace lors de la recherche.
> - Vous pouvez combiner la recherche en texte libre à d'autres options de recherche telles que Document type (Type de document) ou Platform type (Type de plateforme).
> - Le filtre Document Name Prefix (Préfixe de nom de document) et la recherche en texte libre ne peuvent pas être utilisés ensemble. Ils s'excluent mutuellement.

Pour rechercher un document SSM

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.
3. Saisissez vos termes de recherche dans la zone de recherche, puis appuyez sur Entrée.

Effectuer une recherche dans un document texte libre à l'aide du AWS CLI

Pour effectuer une recherche de documents en texte libre en utilisant la CLI

1. Installez et configurez le AWS Command Line Interface (AWS CLI), si ce n'est pas déjà fait.

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l' AWS CLI.
2. Pour effectuer une recherche de document en texte libre avec un seul terme, exécutez la commande suivante. Dans cette commande, remplacez *search_term* par 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 avec plusieurs termes qui créent une instruction AND, exécutez la commande suivante. Dans cette commande, remplacez *search_term_1* et *search_term_2* par vos propres informations.

   ```
   aws ssm list-documents --filters
    Key="SearchKeyword",Values="search_term_1","search_term_2","search_term_3" --region us-east-2
   ```

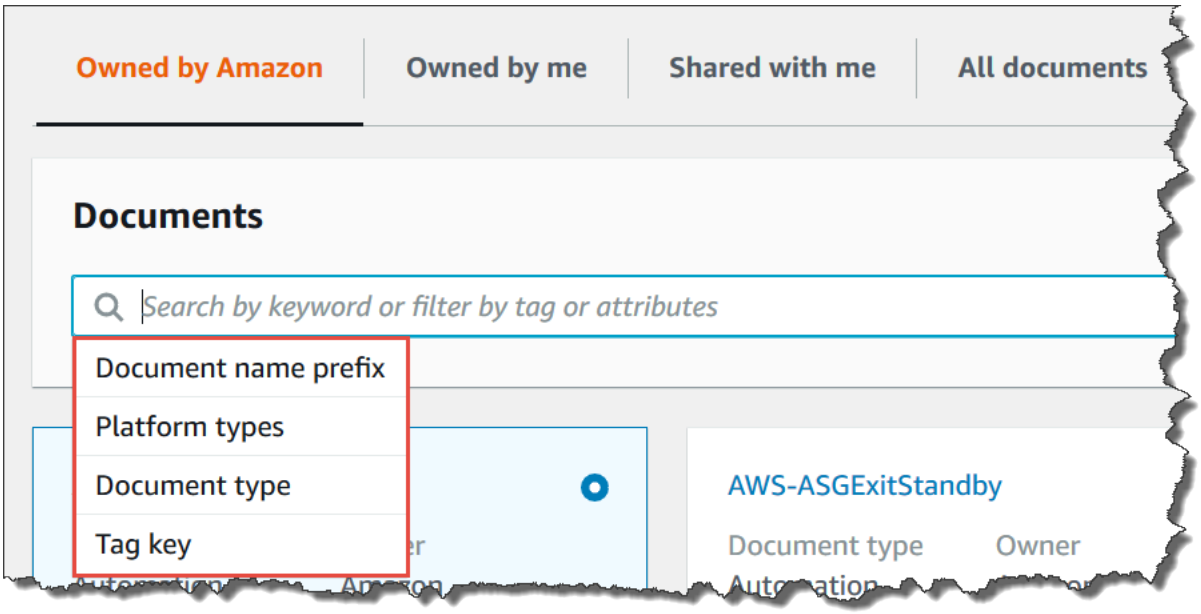
   Voici un exemple :

```
aws ssm list-documents --filters Key="SearchKeyword",Values="aws-asg","aws-
ec2","restart" --region us-east-2
```

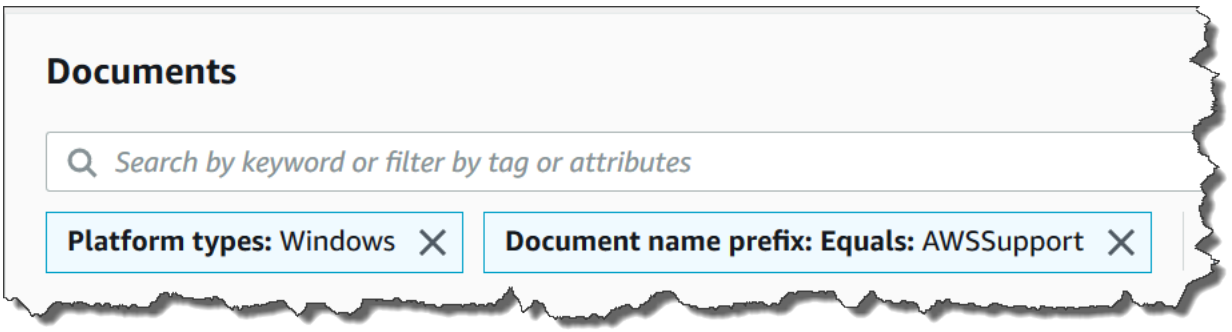
Utilisation de filtres

La page Documents de Systems Manager affiche automatiquement les filtres suivants lorsque vous sélectionnez la zone de recherche.

- Préfixe de nom de document
- Types de plateforme
- Type de document
- Clé de balise



Vous pouvez rechercher des documents SSM en utilisant un seul filtre. Si vous voulez renvoyer un ensemble plus spécifique de documents SSM, vous pouvez appliquer plusieurs filtres. Voici un exemple d'une recherche qui utilise les filtres Types de plateforme et Préfixe de nom de document.



Si vous appliquez plusieurs filtres, Systems Manager crée différentes instructions de recherche en fonction des filtres choisis :

- Si vous appliquez le même filtre plusieurs fois, Préfixe de nom de document par exemple, Systems Manager effectue une recherche en utilisant une instruction `OR`. Par exemple, si vous spécifiez un filtre de Préfixe de nom de document=**`AWS`** et un second filtre de Préfixe de nom de document=**`Lambda`**, la recherche renvoie tous les documents avec le préfixe « `AWS` » et tous les documents avec le préfixe « `Lambda` ».
- Si vous appliquez différents filtres, par exemple, Document name prefix (Préfixe de nom de document) et Platform types, (Types de plateforme), alors Systems Manager effectuera une recherche en utilisant une instruction `AND`. Si vous indiquez par exemple un filtre Document name prefix (Préfixe de nom de document)=**`AWS`** et un filtre Platform types (Types de plateforme)=**`Linux`**, la recherche renverra alors tous les documents avec le préfixe « `AWS` » spécifiques à la plateforme Linux.

> **Note**
>
> Les recherches qui utilisent des filtres sont sensibles à la casse.

Ajouter des documents à vos favoris

Pour vous aider à trouver les documents SSM fréquemment utilisés, ajoutez des documents à vos favoris. Vous pouvez ajouter jusqu'à 20 documents à vos favoris par type de document, par Compte AWS et Région AWS. Vous pouvez choisir, modifier et consulter vos favoris à partir des documents de la AWS Management Console. Les procédures suivantes décrivent comment sélectionner, modifier et afficher vos favoris.

Pour ajouter un document SSM aux favoris, procédez comme suit :

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.
3. Sélectionnez l'icône en forme d'étoile à côté du nom du document que vous souhaitez ajouter à vos favoris.

Pour supprimer un document SSM de vos favoris, procédez comme suit :

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.
3. Désélectionnez l'icône en forme d'étoile à côté du nom du document que vous souhaitez supprimer de vos favoris.

Pour afficher vos favoris à partir des documents AWS Management Console

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.
3. Sélectionnez l'onglet Favoris.

Sécurité dans AWS Systems Manager

Chez Amazon Web Services, la sécurité dans le cloud est la priorité la plus élevée. 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 cela comme la sécurité du cloud et la sécurité dans le cloud :

- Sécurité du cloud – AWS est responsable de la protection de l'infrastructure qui exécute des Services AWS dans le AWS Cloud. 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 AWS programmes de conformité. Pour en savoir plus sur les programmes de conformité qui s'appliquent à AWS Systems Manager, reportez-vous aux Services AWS concernés par le programme de conformité.
- Sécurité dans le cloud – Votre responsabilité est fonction du Service AWS que vous utilisez. Vous êtes également responsable d'autres facteurs, y compris de la sensibilité de vos données, des exigences de votre entreprise,et de la législation et de la réglementation applicables.

Cette documentation vous aide à comprendre comment appliquer le modèle de responsabilité partagée lors de l'utilisation de AWS Systems Manager. Les rubriques suivantes expliquent 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

- Bonnes pratiques de sécurité pour Systems Manager

Protection des données dans AWS Systems Manager

La protection des données fait référence à la protection des données en transit (lors de leur trajet aller-retourSystems Manager) et au repos (lorsqu'elles sont stockées dans AWS des centres de données).

Le modèle de responsabilité AWS partagée de s'applique à la protection des données dansAWS Systems Manager. Comme décrit dans ce modèle, AWS est chargé de protéger l'infrastructure mondiale qui gère tous les AWS Cloud. La gestion du contrôle de votre contenu hébergé sur cette infrastructure relève de votre responsabilité. Vous êtes également responsable des tâches de configuration et de gestion de la sécurité des Services AWS que vous utilisez. Pour plus d'informations sur la confidentialité des données, consultez Questions fréquentes (FAQ) sur la confidentialité des données. Pour en savoir plus sur la protection des données en Europe, consultez le billet de blog Modèle de responsabilité partagée AWS et RGPD (Règlement général sur la protection des données) sur le Blog de sécuritéAWS .

À des fins de protection des données, nous vous recommandons de protéger les Compte AWS informations d'identification et de configurer les utilisateurs individuels avec AWS IAM Identity Center ou AWS Identity and Access Management (IAM). Ainsi, chaque utilisateur se voit attribuer uniquement les autorisations nécessaires pour exécuter ses tâches. Nous vous recommandons également de sécuriser vos données comme indiqué ci-dessous :

- Utilisez l'authentification multifactorielle (MFA) avec chaque compte.
- Utilisez le protocole SSL/TLS pour communiquer avec les ressources. AWS Nous exigeons TLS 1.2 et recommandons TLS 1.3.
- Configurez l'API et la journalisation de l'activité des utilisateurs avec AWS CloudTrail.
- Utilisez des solutions de AWS chiffrement, ainsi que tous les contrôles de sécurité par défaut qu'ils contiennent Services AWS.
- Utilisez des services de sécurité gérés avancés tels qu'Amazon Macie, qui contribuent à la découverte et à la sécurisation des données sensibles stockées dans Amazon S3.
- Si vous avez besoin de modules cryptographiques validés par la norme FIPS 140-2 pour accéder AWS via une interface de ligne de commande ou une API, utilisez un point de terminaison FIPS. Pour plus d'informations sur les points de terminaison FIPS (Federal Information Processing

Standard) disponibles, consultez Federal Information Processing Standard (FIPS) 140-2 (Normes de traitement de l'information fédérale).

Nous vous recommandons fortement de ne jamais placer d'informations confidentielles ou sensibles, telles que les adresses e-mail de vos clients, dans des balises ou des champs de texte libre tels que le champ Name (Nom). Cela inclut lorsque vous travaillez avec Systems Manager ou d'autres Services AWS utilisateurs de la console, de l'API ou AWS des SDK. AWS CLI Toutes les données que vous entrez dans des balises ou des champs de texte de forme libre utilisés pour les noms peuvent être utilisées à des fins de facturation ou dans les journaux de diagnostic. Si vous fournissez une adresse URL à un serveur externe, nous vous recommandons fortement de ne pas inclure d'informations d'identification dans l'adresse URL permettant de valider votre demande adressée à ce serveur.

Chiffrement des données

Chiffrement au repos

Parameter Store paramètres

Vous pouvez créer des paramètres du type `String`, `StringList` et `SecureString` dans Parameter Store, une fonctionnalité de AWS Systems Manager.

Pour chiffrer les valeurs des `SecureString` paramètres, Parameter Store utilise un AWS KMS key in AWS Key Management Service (AWS KMS). AWS KMS utilise soit une clé gérée par le client, soit un Clé gérée par AWS pour chiffrer la valeur du paramètre dans une base de données AWS gérée.

> **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 Qu'est-ce qu'un paramètre ? et Restriction de l'accès aux paramètres Systems Manager à l'aide des politiques IAM.

Contenu des compartiments 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 de compartiment S3, consultez Protection des données à l'aide du chiffrement et Protection des données dans Amazon S3 dans le Guide de l'utilisateur Amazon Simple Storage Service.

Voici les types de données que vous pouvez charger ou stocker dans les compartiments S3 dans le cadre de vos activités Systems Manager :

- La sortie des commandes en entréeRun Command, une capacité de AWS Systems Manager
- Emballé dansDistributor, une capacité de AWS Systems Manager
- L'opération de correction permet de se connecterPatch Manager, une fonctionnalité de AWS Systems Manager
- Listes de remplacement de correctifs Patch Manager
- Scripts ou Ansible Playbooks à exécuter dans un flux de travail Runbook dans Automation, une fonctionnalité de AWS Systems Manager
- Chef InSpecprofils à utiliser avec les scans dans Compliance, une fonctionnalité de AWS Systems Manager
- AWS CloudTrail journaux
- L'historique des sessions permet de se Session Manager connecter, une fonctionnalité de AWS Systems Manager
- Des rapports provenant deExplorer, une capacité de AWS Systems Manager
- OpsData à partir deOpsCenter, une capacité de AWS Systems Manager
- AWS CloudFormation modèles à utiliser avec les flux de travail d'automatisation
- Données de conformité issues d'une analyse de synchronisation de données de ressources
- Sortie de demandes de création ou de modification d'une association dansState Manager, une fonctionnalité de AWS Systems Manager, sur des nœuds gérés
- Documents Systems Manager (documents SSM) personnalisés, que vous pouvez exécuter en utilisant le document SSM AWS géré par `AWS-RunDocument`

CloudWatch Logs, journaux, groupes

Dans le cadre de vos Systems Manager opérations, vous pouvez choisir de diffuser des données vers un ou plusieurs groupes de CloudWatch journaux Amazon Logs.

Pour plus d'informations sur le chiffrement des groupes de CloudWatch journaux, consultez la section Chiffrer les données des CloudWatch journaux dans les journaux à l'aide AWS Key Management Service du guide de l'utilisateur Amazon CloudWatch Logs.

Les types de données que vous avez peut-être transmis à un groupe de CloudWatch journaux Logs dans le cadre de vos Systems Manager activités sont les suivants :

- Sortie des commandes Run Command
- Sortie des scripts exécutés à l'aide de l'action `aws:executeScript` dans un runbook Automation
- Journaux d'historique de session Session Manager
- Journaux provenant de l'SSM Agent sur vos nœuds gérés

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 nœuds.

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 Systems Manager des ressources à l'aide du AWS Management Console AWS SDK ou de l'Systems ManagerAPI, toutes les communications sont cryptées avec le protocole TLS (Transport Layer Security). Pour obtenir la liste complète des points de terminaison d'API, veuillez consulter la rubrique Points de terminaison de Service AWS de la Référence générale d'Amazon Web Services.

Instances gérées

AWS fournit une connectivité sécurisée et privée entre les instances Amazon Elastic Compute Cloud (Amazon EC2). De plus, nous chiffrons automatiquement le trafic en transit entre les instances prises en charge dans le même Virtual Private Cloud (VPC) ou dans des VPC appairés à l'aide des algorithmes AEAD avec un 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 AWS KMS key d'un code créé dans AWS KMS. AWS KMS le chiffrement est disponible pour `Standard_StreamInteractiveCommands`, et les types de `NonInteractiveCommands` session.

Accès via Run Command

Par défaut, l'accès distant à vos nœuds via Run Command est chiffré à l'aide de TLS 1.2 et les demandes de création d'une connexion sont chiffrées à l'aide de SigV4.

Confidentialité du trafic inter-réseau

Vous pouvez utiliser Amazon Virtual Private Cloud (Amazon VPC) pour créer des limites entre les ressources de vos nœuds gérés et contrôler le trafic entre ceux-ci, votre réseau sur site et Internet. Pour plus de détails, consultez Améliorer la sécurité des instances EC2 en utilisant des points de terminaison VPC pour Systems Manager.

Pour plus d'informations sur la sécurité Amazon Virtual Private Cloud, consultez Confidentialité du trafic inter-réseau dans Amazon VPC dans le Guide de l'utilisateur Amazon VPC.

Gestion des identités et des accès pour AWS Systems Manager

AWS Identity and Access Management (IAM) est un Service AWS qui aide un administrateur à contrôler en toute sécurité l'accès aux ressources AWS. Des administrateurs IAM contrôlent les personnes qui s'authentifient (sont connectées) et sont autorisées (disposent d'autorisations) à utiliser des ressources Systems Manager. IAM est un Service AWS que vous pouvez utiliser sans frais supplémentaires.

Rubriques

- AWS politiques gérées pour AWS Systems Manager
- Résolution des problèmes d'identité et d'accès avec AWS Systems Manager

Public ciblé

Votre utilisation d'AWS Identity and Access Management (IAM) diffère selon la tâche que vous accomplissez 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 fonctions Systems Manager pour effectuer votre travail, plus vous pourriez avoir besoin d'autorisations supplémentaires. En comprenant bien la gestion des accès, vous saurez demander les autorisations appropriées à votre administrateur. Si vous ne pouvez pas accéder à une fonctionnalité dans Systems Manager, consultez Résolution des problèmes d'identité et d'accès avec AWS Systems Manager.

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. Votre responsabilité est de déterminer Systems Manager les fonctionnalités ainsi que les ressources auxquelles les utilisateurs de votre service doivent 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, veuillez consulter Fonctionnement d'AWS Systems Manager avec IAM.

Administrateur IAM : si vous êtes un administrateur IAM, vous souhaiterez peut-être en savoir plus sur la façon d'écrire des politiques pour gérer l'accès à Systems Manager. Pour voir des exemples de politiques Systems Manager basées sur l'identité que vous pouvez utiliser dans IAM, veuillez consulter Exemples de politiques basées sur l'identité AWS Systems Manager.

Authentification par des identités

L'authentification correspond au processus par lequel vous vous connectez à AWS avec vos informations d'identification. Vous devez vous authentifier (être connecté à AWS) en tant qu'Utilisateur racine d'un compte AWS, en tant qu'utilisateur IAM ou en endossant un rôle IAM.

Vous pouvez vous connecter à AWS en tant qu'identité fédérée à l'aide des informations d'identification fournies par le biais d'une source d'identité. AWS IAM Identity Center Les utilisateurs (IAM Identity Center), l'authentification de connexion unique de votre entreprise et vos informations

d'identification Google ou Facebook sont des exemples d'identités fédérées. Lorsque vous vous connectez avec une identité fédérée, votre administrateur aura précédemment configuré une fédération d'identités avec des rôles IAM. Lorsque vous accédez à AWS en utilisant la fédération, vous endossez indirectement un rôle.

Selon le type d'utilisateur que vous êtes, vous pouvez vous connecter à la AWS Management Console ou au portail d'accès AWS. Pour plus d'informations sur la connexion à AWS, consultez Connexion à votre Compte AWSdans le Guide de l'utilisateurConnexion à AWS.

Si vous accédez à AWS par programmation, AWS fournit un kit de développement logiciel (SDK) et une interface de ligne de commande (CLI) pour signer cryptographiquement vos demandes en utilisant vos informations d'identification. Si vous n'utilisez pas les outils AWS, vous devez signer les requêtes vous-même. Pour plus d'informations sur l'utilisation de la méthode recommandée pour signer des demandes vous-même, consultez Signature des demandes d'API AWS dans le Guide de l'utilisateur IAM.

Quelle que soit la méthode d'authentification que vous utilisez, vous devrez peut-être fournir des informations de sécurité supplémentaires. Par exemple, AWS vous recommande d'utiliser l'authentification multifactorielle (MFA) pour améliorer la sécurité de votre compte. Pour en savoir plus, veuillez consulter Authentification multifactorielle dans le Guide de l'utilisateur AWS IAM Identity Center et Utilisation de l'authentification multifactorielle (MFA) dans l'interface AWS dans le Guide de l'utilisateur IAM.

Utilisateur root 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 AWS et ressources du compte. Cette identité est appelée utilisateur root du Compte AWS. Vous pouvez y accéder en vous connectant à l'aide de l'adresse électronique et du mot de passe que vous avez utilisés pour créer le compte. Il est vivement recommandé de ne pas utiliser l'utilisateur root pour vos tâches quotidiennes. Protégez vos informations d'identification d'utilisateur root et utilisez-les pour effectuer les tâches que seul l'utilisateur root peut effectuer. Pour obtenir la liste complète des tâches qui vous imposent de vous connecter en tant qu'utilisateur root, consultez Tâches nécessitant les informations d'identification de l'utilisateur root dans le Guide de l'utilisateur IAM.

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. Dans la mesure du possible, nous vous recommandons de

vous appuyer sur des informations d'identification temporaires plutôt que de créer des utilisateurs IAM ayant des informations d'identification à long terme tels que les clés d'accès. Toutefois, si certains cas d'utilisation spécifiques nécessitent des informations d'identification à long terme avec les utilisateurs IAM, nous vous recommandons de faire pivoter les clés d'accès. Pour plus d'informations, consultez Rotation régulière des clés d'accès pour les cas d'utilisation nécessitant des informations d'identification dans le Guide de l'utilisateur IAM.

Un groupe IAM est une identité qui concerne 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 IAMAdmins et accorder à ce groupe les autorisations 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 en savoir plus, consultez Quand créer un utilisateur IAM (au lieu d'un rôle) dans le Guide de l'utilisateur IAM.

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 le rôle IAM n'est pas associé à une personne en particulier. Vous pouvez temporairement endosser un rôle IAM dans la AWS Management Console en changeant 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 Guide de l'utilisateur IAM.

Les rôles IAM avec des informations d'identification temporaires sont utiles dans les cas suivants :

- Accès utilisateur fédéré – Pour attribuer des autorisations à une identité fédérée, vous créez un rôle et définissez des autorisations pour le rôle. Quand une identité externe s'authentifie, l'identité est associée au rôle et reçoit les autorisations qui sont définies par celui-ci. Pour obtenir des informations sur les rôles pour la fédération, consultez Création d'un rôle pour un fournisseur d'identité tiers (fédération) dans le Guide de l'utilisateur IAM. Si vous utilisez IAM Identity Center, vous configurez un jeu d'autorisations. IAM Identity Center met en corrélation le jeu d'autorisations avec un rôle dans IAM afin de contrôler à quoi vos identités peuvent accéder après leur authentification. Pour plus d'informations sur les jeux d'autorisations, consultez Jeux d'autorisations dans le Guide de l'utilisateur AWS IAM Identity Center.

- Autorisations d'utilisateur IAM temporaires : un rôle ou un utilisateur IAM peut endosser un rôle IAM pour profiter temporairement d'autorisations différentes pour une tâche spécifique.
- Accès intercompte : vous pouvez utiliser un rôle IAM pour permettre à un utilisateur (principal de confiance) d'un compte différent d'accéder aux ressources de votre compte. Les rôles constituent le principal moyen d'accorder l'accès intercompte. Toutefois, certains Services AWS vous permettent d'attacher une politique 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 politiques basées sur les ressources pour l'accès intercompte, consultez Différence entre les rôles IAM et les politiques basées sur les ressources dans le Guide de l'utilisateur IAM.
- Accès interservices : certains Services AWS utilisent des fonctions dans d'autres Services AWS. Par exemple, lorsque vous effectuez un appel dans un service, il est courant que ce service exécute des applications dans Amazon EC2 ou stocke des objets dans Amazon S3. Un service peut le faire en utilisant les autorisations d'appel du principal, une fonction du service ou un rôle lié au service.
  - Forward access sessions (FAS) – Lorsque vous utilisez un utilisateur ou un rôle IAM pour effectuer des actions dans AWS, vous êtes considéré comme un principal. Lorsque vous utilisez certains services, vous pouvez effectuer une action qui déclenche une autre action dans un autre service. FAS utilise les autorisations du principal appelant Service AWS, combinées à la demande Service AWS pour effectuer des demandes aux services en aval. Les demandes de FAS ne sont effectuées que lorsqu'un service reçoit une demande qui nécessite des interactions avec d'autres Services AWS ou des ressources pour être traitée. Dans ce cas, vous devez disposer d'autorisations nécessaires pour effectuer les deux actions. Pour plus de détails sur la politique relative à la transmission de demandes FAS, consultez Sessions de transmission d'accès.
  - Fonction du service : il s'agit d'un rôle IAM attribué à un service afin de réaliser des actions en votre nom. Un administrateur IAM peut créer, modifier et supprimer une fonction du service à partir d'IAM. Pour plus d'informations, consultez Création d'un rôle pour la délégation d'autorisations à un Service AWS dans le Guide de l'utilisateur IAM.
  - Rôle lié au service – Un rôle lié au service est un type de fonction du service lié à un Service AWS. Le service peut endosser le rôle afin d'effectuer une action en votre nom. Les rôles liés à un service s'affichent dans votre Compte AWS 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.
- Applications s'exécutant sur Amazon EC2 : vous pouvez utiliser un rôle IAM pour gérer des informations d'identification temporaires pour les applications s'exécutant sur une instance EC2

et effectuant 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 Guide de l'utilisateur IAM.

Pour savoir dans quel cas utiliser des rôles ou des utilisateurs IAM, consultez Quand créer un rôle IAM (au lieu d'un utilisateur) dans le Guide de l'utilisateur IAM.

Gestion des accès à l'aide de politiques

Vous contrôlez les accès dans AWS en créant des politiques et en les attachant à des identités AWS ou à des ressources. Une politique 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 politiques lorsqu'un principal (utilisateur, utilisateur racine ou séance de rôle) envoie une demande. Les autorisations dans les politiques déterminent si la demande est autorisée ou refusée. La plupart des politiques sont stockées dans AWS en tant que documents JSON. Pour plus d'informations sur la structure et le contenu des documents de politique JSON, consultez Présentation des politiques JSON dans le Guide de l'utilisateur IAM.

Les administrateurs peuvent utiliser les politiques JSON AWS pour spécifier qui a accès à quoi. C'est-à-dire, quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

Par défaut, les utilisateurs et les rôles ne disposent d'aucune autorisation. Pour octroyer aux utilisateurs des autorisations d'effectuer des actions sur les ressources dont ils ont besoin, un administrateur IAM peut créer des politiques IAM. L'administrateur peut ensuite ajouter les politiques IAM aux rôles et les utilisateurs peuvent assumer les rôles.

Les politiques 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 politique qui autorise l'action `iam:GetRole`. Un utilisateur avec cette politique peut obtenir des informations utilisateur à partir de la AWS Management Console, de la AWS CLI ou de l'API AWS.

Politiques basées sur l'identité

Les politiques basées sur l'identité sont des documents de politique d'autorisations JSON que vous pouvez attacher à une identité telle qu'un utilisateur, un groupe d'utilisateurs ou un rôle IAM. Ces politiques contrôlent quel type d'actions des utilisateurs et des rôles peuvent exécuter, sur quelles ressources et dans quelles conditions. Pour découvrir comment créer une politique basée sur l'identité, consultez Création de politiques IAM dans le Guide de l'utilisateur IAM.

Les politiques basées sur l'identité peuvent être classées comme des politiques en ligne ou des politiques gérées. Les politiques en ligne sont intégrées directement à un utilisateur, groupe ou rôle. Les politiques gérées sont des politiques autonomes que vous pouvez attacher à plusieurs utilisateurs, groupes et rôles dans votre Compte AWS. Les politiques gérées incluent les politiques gérées par AWS et les politiques gérées par le client. Pour découvrir comment choisir entre une politique gérée et une politique en ligne, consultez Choix entre les politiques gérées et les politiques en ligne dans le Guide de l'utilisateur IAM.

Pour plus d'informations sur les politiques gérées AWS pour Systems Manager, veuillez consulter Politiques gérées par AWS Systems Manager.

Politiques basées sur les ressources

Les politiques basées sur les ressources sont des documents de politique JSON que vous attachez à une ressource. Des politiques basées sur les ressources sont, par exemple, les politiques de confiance de rôle IAM et des politiques de compartiment Amazon S3. Dans les services qui sont compatibles avec les politiques basées sur les ressources, les administrateurs de service peuvent les utiliser pour contrôler l'accès à une ressource spécifique. Pour la ressource dans laquelle se trouve la politique, cette dernière définit quel type d'actions un principal spécifié peut effectuer sur cette ressource et dans quelles conditions. Vous devez spécifier un principal dans une politique basée sur les ressources. Les principaux peuvent inclure des comptes, des utilisateurs, des rôles, des utilisateurs fédérés ou des Services AWS.

Les politiques basées sur les ressources sont des politiques en ligne situées dans ce service. Vous ne pouvez pas utiliser les politiques gérées AWS depuis IAM dans une politique basée sur une ressource.

Listes de contrôle d'accès (ACL)

Les listes de contrôle d'accès (ACL) vérifie quels principaux (membres de compte, utilisateurs ou rôles) ont l'autorisation d'accéder à une ressource. Les listes de contrôle d'accès sont similaires aux

politiques basées sur les ressources, bien qu'elles n'utilisent pas le format de document de politique JSON.

Amazon S3, AWS WAF et Amazon VPC sont des exemples de services prenant en charge les ACL. Pour en savoir plus sur les listes de contrôle d'accès, consultez Présentation des listes de contrôle d'accès (ACL) dans le Guide du développeur Amazon Simple Storage Service.

Autres types de politique

AWS prend en charge d'autres types de politiques moins courantes. Ces types de politiques peuvent définir le nombre maximum d'autorisations qui vous sont accordées par des types de politiques plus courants.

- Limite d'autorisations : une limite d'autorisations est une fonction avancée dans laquelle vous définissez le nombre maximal d'autorisations qu'une politique 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 qui en résultent représentent la combinaison des politiques basées sur l'identité d'une entité et de ses limites d'autorisation. Les politiques 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 politiques remplace l'autorisation. Pour plus d'informations sur les limites d'autorisations, consultez Limites d'autorisations pour des entités IAM dans le Guide de l'utilisateur IAM.
- Politiques de contrôle des services (SCP) - les SCP sont des politiques 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 politiques de contrôle des services (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 SCP dans le Guide de l'utilisateur AWS Organizations.
- politiques de séance : les politiques de séance sont des politiques avancées que vous utilisez en tant que paramètre lorsque vous créez par programmation une séance temporaire pour un rôle ou un utilisateur fédéré. Les autorisations de la séance obtenue sont une combinaison des politiques basées sur l'identité de l'utilisateur ou du rôle et des politiques de séance. Les autorisations peuvent également provenir d'une politique basée sur les ressources. Un refus explicite dans l'une de ces politiques remplace l'autorisation. Pour plus d'informations, consultez Politiques de séance dans le Guide de l'utilisateur IAM.

Plusieurs types de politique

Lorsque plusieurs types de politiques 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 politiques, veuillez consulter Logique d'évaluation de politiques dans le Guide de l'utilisateur IAM.

Fonctionnement d'AWS Systems Manager avec IAM

Avant d'utiliser AWS Identity and Access Management (IAM) pour gérer l'accès à AWS Systems Manager, vous devez connaître les fonctionnalités IAM disponibles. Systems Manager Pour obtenir une vue d'ensemble de la manière dont vous Services AWS travaillez avec IAM Systems Manager et des autres méthodes utilisées, consultez Services AWS le guide de l'utilisateur d'IAM consacré à l'utilisation d'IAM.

Rubriques

- Systems ManagerPolitiques basées sur l'identité
- Systems ManagerPolitiques basées sur les ressources
- Autorisation basée sur les balises Systems Manager
- Rôles IAM Systems Manager

Systems ManagerPolitiques basées sur l'identité

Avec les politiques basées sur l'identité IAM, vous pouvez spécifier les actions et les ressources autorisées ou refusées, ainsi que les conditions dans lesquelles les actions sont autorisées ou refusées. Systems Manager prend en charge des actions, des ressources et des clés de condition spécifiques. Pour en savoir plus sur tous les éléments que vous utilisez dans une politique JSON, consultez Références des éléments de politique JSON IAM dans le Guide de l'utilisateur IAM.

Actions

Les administrateurs peuvent utiliser les politiques AWS JSON pour spécifier qui a accès à quoi. C'est-à-dire, quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

L'élément `Action` d'une politique JSON décrit les actions que vous pouvez utiliser pour autoriser ou refuser l'accès à une politique. Les actions de stratégie portent généralement le même nom

que l'opération AWS d'API associée. Il existe quelques exceptions, telles que les actions avec autorisations uniquement qui n'ont pas d'opération API correspondante. Certaines opérations nécessitent également plusieurs actions dans une politique. Ces actions supplémentaires sont nommées actions dépendantes.

Intégration d'actions dans une politique afin d'accorder l'autorisation d'exécuter les opérations associées.

Les actions de politique dans Systems Manager utilisent le préfixe suivant avant l'action : `ssm:`. Par exemple, pour accorder à une personne l'autorisation de créer un paramètre Systems Manager (paramètre SSM) à l'aide de l'opération d'API Systems Manager `PutParameter`, vous incluez l'action `ssm:PutParameter` dans sa politique. Les déclarations de politique 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"
```

> **Note**
>
> Les fonctionnalités suivantes permettent d' AWS Systems Manager utiliser différents préfixes avant les actions.
>
> - AWS AppConfig utilise le préfixe `appconfig:` avant les actions.
> - Incident Manager utilise le préfixe `ssm-incidents:` ou `ssm-contacts:` avant les actions.
> - Systems Manager GUI Connect utilise le préfixe `ssm-guiconnect` avant les actions.

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, consultez Actions définies par AWS Systems Manager dans la Référence de l'autorisation de service.

Ressources

Les administrateurs peuvent utiliser les politiques AWS JSON pour spécifier qui a accès à quoi. C'est-à-dire, quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

L'élément de politique JSON `Resource` indique le ou les objets auxquels l'action s'applique. Les instructions doivent inclure un élément `Resource` ou `NotResource`. Il est recommandé de définir une ressource à l'aide de son Amazon Resource Name (ARN). Vous pouvez le faire pour des actions qui prennent en charge un type de ressource spécifique, connu sous la dénomination autorisations de niveau ressource.

Pour les actions qui ne sont pas compatibles avec les autorisations de niveau ressource, telles que les opérations de liste, utilisez un caractère générique (*) afin d'indiquer que l'instruction s'applique à toutes les ressources.

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

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 déclaration dans la Région USA Est (Ohio), vous utiliserez 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:region:123456789012:maintenancewindow/*"
```

Pour les opérations d'`Parameter Store`API, vous pouvez fournir ou restreindre l'accès à tous les paramètres d'un niveau d'une hiérarchie en utilisant des noms hiérarchiques et des politiques AWS Identity and Access Management (IAM) comme suit.

```
"Resource": "arn:aws:ssm:region:123456789012:parameter/Dev/ERP/Oracle/*"
```

Certaines actions Systems Manager, telles que la création de ressources, ne peuvent pas être exécutées sur une ressource spécifique. Dans ces cas-là, vous devez utiliser le caractère générique (*).

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

Certaines opérations d'API Systems Manager acceptent plusieurs ressources. Pour spécifier plusieurs ressources dans une seule instruction, séparez leurs ARN par des virgules, comme suit.

```
"Resource": [
      "resource1",
      "resource2"
```

> **Note**
>
> La plupart Services AWS traitent deux points (:)) ou une barre oblique (/) comme le même caractère dans les ARN. 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 ci-dessous décrit les formats ARN pour les types de ressources pris en charge parSystems Manager.

> **Note**
>
> Notez les exceptions suivantes relatives aux formats ARN.
>
> - Les fonctionnalités suivantes permettent d' AWS Systems Manager utiliser différents préfixes avant les actions.
>   - AWS AppConfig utilise le préfixe `appconfig:` avant les actions.
>   - Incident Manager utilise le préfixe `ssm-incidents:` ou `ssm-contacts:` avant les actions.
>   - Systems Manager GUI Connect utilise le préfixe `ssm-guiconnect` avant les actions.

- Les documents et les ressources de définition d'automatisation détenus par Amazon, ainsi que les paramètres publics fournis par Amazon et par des sources tierces, n'incluent pas les identifiants de compte dans leurs formats ARN. Par exemple :
  - Le document `AWS-RunPatchBaseline` SSM :

    ```
    arn:aws:ssm:us-east-2:::document/AWS-RunPatchBaseline
    ```

  - Le manuel d'automatisation `AWS-ConfigureMaintenanceWindows` :

    ```
    arn:aws:ssm:us-east-2:::automation-definition/AWS-
    ConfigureMaintenanceWindows
    ```

  - Le paramètre public `/aws/service/bottlerocket/aws-ecs-1-nvidia/x86_64/1.13.4/image_version` :

    ```
    arn:aws:ssm:us-east-2::parameter/aws/service/bottlerocket/aws-
    ecs-1-nvidia/x86_64/1.13.4/image_version
    ```

  Pour plus d'informations sur ces trois types de ressources, consultez les rubriques suivantes :

  - Utilisation de documents
  - Exécution d'automatisations
  - Utilisation de paramètres publics

Type de ressource	Format ARN
Application (AWS AppConfig)	arn:aws:appconfig:*region*:*account-id* :application/*application-id*
Association	arn:aws:ssm:*region*:*account-id* :association/*association-id*
Exécution d'Automation	arn:aws:ssm:*region*:*account-id* :automation-execution/*automation-execution-id*
Définition d'Automation (avec sous-ressource de version)	arn:aws:ssm:*region*:*account-id* :automation-definition/*automation-definition-id* :*version-id* [1]

Type de ressource	Format ARN
Profil de configuration (AWS AppConfig)	arn:aws:appconfig:*region*:*account-id* :application/*application-id* /configurationprofile/*configurationprofile-id*
Contact (Incident Manager)	arn:aws:ssm-contacts:*region*:*account-id* :contact/*contact-alias*
Politique de déploiement (AWS AppConfig)	arn:aws:appconfig:*region*:*account-id* :deploymentstrategy/*deploymentstrategy-id*
Document	arn:aws:ssm:*region*:*account-id* :document/*document-name*
Environnement (AWS AppConfig)	arn:aws:appconfig:*region*:*account-id* :application/*application-id* /environment/*environment-id*
Incident	arn:aws:ssm-incidents:*region*:*account-id* :incident-record/*response-plan-name* /*incident-id*
Fenêtre de maintenance	arn:aws:ssm:*region*:*account-id* :maintenancewindow/*window-id*
Nœud géré	arn:aws:ssm:*region*:*account-id* :managed-instance/*managed-node-id*
Inventaire des nœuds gérés	arn:aws:ssm:*region*:*account-id* :managed-instance-inventory /*managed-node-id*
OpsItem	*arn:aws:ssm : region : account-id : opsitem/ -id OpsItem*

Type de ressource	Format ARN
Paramètre	Un paramètre de premier niveau : • arn:aws:ssm:*region*:*account-id* :parameter/*parameter-name*/ Un paramètre nommé avec une construction hiérarchique : • arn:aws:ssm:*region*:*account-id* :parameter/*parameter-name-root* /*level-2*/*level-3*/*level-4*/*level-5* [2]
Référentiel de correctifs	arn:aws:ssm:*region*:*account-id* :patchbaseline/*patch-baseline-id*
Plan de réponse	arn:aws:ssm-incidents:*region*:*account-id:* response-plan/*response-plan-name*
Session	arn:aws:ssm:*region*:*account-id* :session/*session-id* [3]
Toutes les ressources Systems Manager	arn:aws:ssm:*
Toutes les Systems Manager ressources détenues par les personnes spécifiées Compte AWS dans les Région AWS	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 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 nœuds.

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 Création de paramètres Systems Manager.

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. Par 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 de plus amples informations sur le format des ARN, veuillez consulter Amazon Ressource Names (ARN) dans la Référence générale d'Amazon Web Services.

Pour afficher la liste des types de ressources Systems Manager, consultez Ressources définies par AWS Systems Manager dans la Référence de l'autorisation de service. Pour savoir grâce à quelles actions vous pouvez spécifier l'ARN de chaque ressource, consultez Actions définies par AWS Systems Manager.

Clés de condition pour Systems Manager

Les administrateurs peuvent utiliser les politiques AWS JSON pour spécifier qui a accès à quoi. C'est-à-dire, quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

L'élément `Condition` (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, tels que les signes égal ou inférieur à, pour faire correspondre la condition de la politique 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 `OR` opération 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 plus d'informations, consultez Éléments d'une politique IAM : variables et identifications dans le Guide de l'utilisateur IAM.

AWS prend en charge les clés de condition globales et les clés de condition spécifiques au service. Pour voir toutes les clés de condition AWS globales, voir les clés de contexte de condition AWS globales dans le guide de l'utilisateur IAM.

Pour afficher la liste des clés de condition Systems Manager, consultez Clés de condition pour AWS Systems Manager dans la Référence de l'autorisation de service. 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/*`, consultez les rubriques suivantes :

- Limitation de l'accès aux commandes de niveau racine via l'SSM Agent
- Restriction de l'accès Run Command en fonction des balises
- Limiter l'accès à la session en fonction des balises d'instance

Pour obtenir des informations sur l'utilisation des clés de condition `ssm:Recursive` et `ssm:Overwrite`, consultez Utiliser des hiérarchies de paramètres.

Exemples

Pour voir des exemples de politiques Systems Manager basées sur l'identité, consultez Exemples de politiques basées sur l'identité AWS Systems Manager.

Systems ManagerPolitiques basées sur les ressources

D'autres Services AWS, comme Amazon Simple Storage Service (Amazon S3), prennent en charge les politiques d'autorisation basées sur les ressources. Par exemple, vous pouvez attacher une politique d'autorisation à un compartiment S3 pour gérer les autorisations d'accès à ce compartiment.

Systems Manager ne prend pas en charge les politiques basées sur une ressource.

Autorisation basée sur les balises Systems Manager

Vous pouvez attacher des balises aux ressources de Systems Manager, ou transmettre des balises dans une demande 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 politique 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
- Nœud géré
- Fenêtre de maintenance
- Paramètre
- Référentiel de correctifs
- OpsItem

Pour plus d'informations sur le balisage des ressources Systems Manager, consultez Balisage des ressources Systems Manager.

Pour visualiser un exemple de politique basée sur l'identité permettant de limiter l'accès à une ressource en fonction des balises de cette ressource, consultez Affichage de documents Systems Manager basés sur des balises.

Rôles IAM Systems Manager

Un rôle IAM est une entité au sein de vous Compte AWS qui possède des autorisations spécifiques.

Utilisation des informations d'identification temporaires avec Systems Manager

Vous pouvez utiliser des informations d'identification temporaires pour vous connecter à l'aide de la fédération, endosser un rôle IAM ou encore pour endosser un rôle intercompte. Vous obtenez des

informations d'identification de sécurité temporaires en appelant AWS Security Token Service (AWS STS) des opérations d'API telles que AssumeRoleou 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 Services AWS d'accéder aux ressources d'autres services pour effectuer une action en votre nom. Les rôles liés à un service s'affichent dans votre compte IAM et sont la propriété du service. Un administrateur 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 plus d'informations sur la création ou la gestion des rôles liés à un service Systems Manager, consultez Utilisation des rôles liés aux services pour Systems Manager.

Rôles de service

Cette fonction permet à un service d'endosser une fonction du 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 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 nœuds gérés, vous devez choisir un rôle autorisant Systems Manager à accéder aux nœuds 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 nœuds gérés.

Pour accéder aux instances EC2, vous devez configurer les autorisations d'instance. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager.

Pour accéder à des nœuds non EC2 dans un environnement hybride et multicloud, votre Compte AWS a besoin d'une fonction du service IAM. Pour plus d'informations, voir Créer le rôle de service IAM requis pour Systems Manager dans les environnements hybrides et multicloud.

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 Configuration d'un accès à un rôle de service (rôle de responsable) pour les automatisations.

Politiques gérées par AWS Systems Manager

AWS répond à de nombreux cas d'utilisation courants en fournissant des politiques IAM autonomes créées et administrées par. AWS Ces politiques gérées AWS octroient les autorisations requises dans les cas d'utilisation courants, ce qui vous évite d'avoir à vous en soucier. (Vous pouvez également créer vos propres politiques IAM personnalisées afin d'accorder des autorisations pour les actions et les ressources Systems Manager.)

Pour plus d'informations sur les politiques gérées pour Systems Manager, voir AWS politiques gérées pour AWS Systems Manager

Pour obtenir des informations générales sur les politiques gérées, voir les politiques AWS gérées dans le guide de l'utilisateur IAM.

Exemples de politiques basées sur l'identité AWS Systems Manager

Par défaut, les utilisateurs et les rôles AWS Identity and Access Management (IAM) ne sont pas autorisés à créer ni à modifier les ressources AWS Systems Manager. Ils ne peuvent pas non plus exécuter de tâches à l'aide de la console Systems Manager, de l'AWS Command Line Interface (AWS CLI) ou de l'API AWS. Un administrateur doit créer des politiques 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 ayant besoin de ces autorisations.

L'exemple suivant illustre une politique d'autorisations qui permet à un utilisateur de supprimer les documents dont le nom commence par **`MyDocument-`** dans la Région AWS USA Est (Ohio) (us-east-2).

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

```
      "Action" : [
        "ssm:DeleteDocument"
      ],
      "Resource" : [
        "arn:aws:ssm:us-east-2:111122223333:document/MyDocument-*"
      ]
    }
  ]
}
```

Pour apprendre à créer une politique basée sur l'identité IAM à l'aide de ces exemples de documents de politique JSON, consultez Création de politiques IAM dans le Guide de l'utilisateur IAM.

Rubriques

- Bonnes pratiques en matière de politiques
- Utilisation de la console Systems Manager
- Autorisation accordée aux utilisateurs pour afficher leurs propres autorisations
- Prévention du problème de l'adjoint confus entre services
- Exemples de politiques gérées par le client
- Affichage de documents Systems Manager basés sur des balises

Bonnes pratiques en matière de politiques

Les stratégies basées sur l'identité 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 politiques basées sur l'identité, suivez ces instructions et recommandations :

- Démarrer avec AWS gérées et évoluez vers les autorisations de moindre privilège - Pour commencer à accorder des autorisations à vos utilisateurs et charges de travail, utilisez les politiques gérées AWS qui accordent des autorisations dans de nombreux cas d'utilisation courants. Ils sont disponibles dans votre Compte AWS. Nous vous recommandons de réduire encore les autorisations en définissant des Politiques gérées par le client AWS qui sont spécifiques à vos cas d'utilisation. Pour de plus amples informations, consultez Politiques gérées AWS ou Politiques gérées AWS pour les activités professionnelles dans le Guide de l'utilisateur IAM.
- Accorder les autorisations de moindre privilège - Lorsque vous définissez des autorisations avec des politiques IAM, accordez uniquement les autorisations nécessaires à l'exécution d'une

seule tâche. Pour ce faire, vous définissez les actions qui peuvent être entreprises sur des ressources spécifiques dans des conditions spécifiques, également appelées autorisations de moindre privilège. Pour plus d'informations sur l'utilisation de IAM pour appliquer des autorisations, consultez Politiques et autorisations dans IAM dans le Guide de l'utilisateur IAM.

- Utiliser des conditions dans les politiques IAM pour restreindre davantage l'accès - Vous pouvez ajouter une condition à vos politiques afin de limiter l'accès aux actions et aux ressources. Par exemple, vous pouvez écrire une condition de politique pour spécifier que toutes les demandes doivent être envoyées via SSL. Vous pouvez également utiliser des conditions pour accorder l'accès aux actions de service si elles sont utilisées via un Service AWS spécifique, comme AWS CloudFormation. Pour plus d'informations, consultez Conditions pour éléments de politique JSON IAM dans le Guide de l'utilisateur IAM.
- Utilisez IAM Access Analyzer pour valider vos politiques IAM afin de garantir des autorisations sécurisées et fonctionnelles - IAM Access Analyzer valide les politiques nouvelles et existantes de manière à ce que les politiques IAM respectent le langage de politique IAM (JSON) et les bonnes pratiques IAM. IAM Access Analyzer fournit plus de 100 vérifications de politiques et des recommandations exploitables pour vous aider à créer des politiques sécurisées et fonctionnelles. Pour de plus amples informations, consultez Validation de politique IAM Access Analyzer dans le Guide de l'utilisateur IAM.
- Authentification multifactorielle (MFA) nécessaire : si vous avez un scénario qui nécessite des utilisateurs IAM ou un utilisateur root dans votre Compte AWS, activez l'authentification multifactorielle pour une sécurité renforcée. Pour exiger le MFA lorsque des opérations d'API sont appelées, ajoutez des conditions MFA à vos politiques. Pour de plus amples informations, consultez Configuration de l'accès aux API protégé par MFA dans le Guide de l'utilisateur IAM.

Pour plus d'informations sur les bonnes pratiques dans IAM, consultez Bonnes pratiques de sécurité dans IAM dans le Guide de l'utilisateur IAM.

Utilisation de la console Systems Manager

Pour accéder à la console Systems Manager, vous devez disposer d'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 politique suivante.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:*",
                "ec2:describeInstances",
                "iam:ListRoles"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": "ssm.amazonaws.com"
                }
            }
        }
    ]
}
```

Si vous créez une politique basée sur l'identité qui est plus restrictive que les autorisations minimales requises, la console ne fonctionnera pas comme prévu pour les entités IAM (utilisateurs et rôles) tributaires de cette politique.

Vous n'avez pas besoin d'accorder les autorisations minimales de console pour les utilisateurs qui effectuent des appels uniquement à AWS CLI ou à 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 politique qui permet aux utilisateurs IAM d'afficher les politiques en ligne et gérées attachées à leur identité d'utilisateur. Cette politique inclut les autorisations 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": "*"
        }
    ]
}
```

Prévention du problème de l'adjoint confus entre services

Le problème de l'adjoint confus est un problème de sécurité dans lequel une entité qui n'a pas l'autorisation d'effectuer une action peut contraindre une entité plus privilégiée à effectuer cette action. Dans AWS, l'emprunt d'identité entre services peut entraîner le problème de député confus. L'usurpation d'identité entre services peut se produire lorsqu'un service (le service appelant) appelle un autre service (le service appelé). Le service appelant peut être manipulé et ses autorisations utilisées pour agir sur les ressources d'un autre client auxquelles on ne serait pas autorisé d'accéder autrement. Pour éviter cela, AWS fournit des outils qui vous aident à protéger vos données pour tous les services avec des principaux de service qui ont eu accès aux ressources de votre compte.

Nous vous recommandons d'utiliser les clés de contexte de condition globale `aws:SourceArn` et `aws:SourceAccount` dans les politiques de ressources afin de limiter les autorisations à la ressource octroyées par AWS Systems Manager à un autre service. Si la valeur `aws:SourceArn` ne contient pas l'ID de compte, tel qu'un Amazon Resource Name (ARN) pour un compartiment S3, vous devez utiliser les deux clés de contexte de condition globale pour limiter les autorisations. Si vous utilisez les deux clés de contexte de condition globale et que la valeur `aws:SourceArn` contient l'ID de compte, la valeur `aws:SourceAccount` et le compte dans la valeur `aws:SourceArn` doivent utiliser le même ID de compte lorsqu'ils sont utilisés dans la même instruction de politique. Utilisez `aws:SourceArn` si vous souhaitez qu'une seule ressource soit associée à l'accès entre services. Utilisez `aws:SourceAccount` si vous souhaitez autoriser l'association d'une ressource de ce compte à l'utilisation interservices.

Les sections suivantes fournissent des exemples de politiques pour les fonctionnalités AWS Systems Manager.

Exemple de politique d'activation hybride

Pour les fonctions du service utilisées dans une activation hybride, la valeur de l'`aws:SourceArn` doit correspondre à l'ARN de l'Compte AWS. Veillez à spécifier la Région AWS dans l'ARN où vous avez créé votre activation hybride. Si vous ne connaissez pas l'ARN complet de la ressource ou si vous spécifiez plusieurs ressources, utilisez la clé de contexte de condition globale `aws:SourceArn` avec des caractères génériques (*) pour les parties inconnues de l'ARN. Par exemple, `arn:aws:ssm:*:`*`region`*`:`*`123456789012`*`:*`.

L'exemple suivant illustre l'utilisation des lés de contexte de condition globale `aws:SourceArn` et `aws:SourceAccount` pour Automation afin de prévenir le problème confus des adjoints dans la Région USA Est (Ohio) (us-east-2).

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"",
         "Effect":"Allow",
         "Principal":{
            "Service":"ssm.amazonaws.com"
         },
         "Action":"sts:AssumeRole",
         "Condition":{
            "StringEquals":{
               "aws:SourceAccount":"123456789012"
            },
            "ArnEquals":{
               "aws:SourceArn":"arn:aws:ssm:us-east-2:123456789012:*"
            }
         }
      }
   ]
}
```

Exemple de politique de synchronisation des données de ressources

Systems Manager Inventory, Explorer, et la conformité vous permettent de créer une synchronisation des données de ressources pour centraliser le stockage de vos données d'opérations (OpsData) dans un compartiment Amazon Simple Storage Service central. Si vous souhaitez chiffrer la synchronisation des données de ressource avec AWS Key Management Service (AWS KMS), vous devez créer une nouvelle clé incluant la politique suivante, ou mettre à jour une clé existante et lui ajouter cette politique. Les clés de condition `aws:SourceArn` et `aws:SourceAccount` de cette politique empêchent le problème du député confus. Voici un exemple de politique.

```
{
    "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:us-east-2:123456789012:key/KMS_key_id",
            "Condition": {
                "StringLike": {
                    "aws:SourceAccount": "123456789012"
                },
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:ssm:*:123456789012:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM"
                }
            }
        }
    ]
}
```

> **Note**
>
> L'ARN de l'exemple de politique permet au système de chiffrer OpsData à partir de toutes les sources, sauf AWS Security Hub. Si vous devez chiffrer des données Security Hub, par exemple si vous utilisez Explorer pour collecter des données Security Hub, vous devez associer une politique supplémentaire spécifiant l'ARN suivant : `"aws:SourceArn": "arn:aws:ssm:*:account-id:role/aws-service-role/opsdatasync.ssm.amazonaws.com/AWSServiceRoleForSystemsManagerOpsDataSync"`

Exemples de politiques gérées par le client

Vous pouvez créer des politiques autonomes que vous gérez dans votre propre Compte AWS. Nous les appelons politiques gérées par le client. Vous pouvez attacher ces politiques à plusieurs entités principales dans votre Compte AWS. Lorsque vous attachez une politique à une entité principale, vous accordez à cette dernière les autorisations définies dans la politique. Pour plus d'informations, consultezExemples de politiques gérées par le client dans le Guide de l'utilisateur IAM.

Les exemples suivants de politiques utilisateur accordent des autorisations pour différentes actions Systems Manager. Utilisez-les pour limiter l'accès à Systems Manager pour vos entités IAM (utilisateurs et rôles). Ces politiques fonctionnent lorsque vous exécutez des actions dans l'API Systems Manager, les kits SDK AWS ou l'AWS CLI. Pour les utilisateurs qui utilisent la console,

vous devez accorder des autorisations supplémentaires propres à la console. Pour de plus amples informations, veuillez consulter Utilisation de la console Systems Manager.

> **Note**
>
> Tous les exemples utilisent la région USA Ouest (Oregon) (us-west-2) et contiennent des ID de compte fictifs. Il est inutile de spécifier l'ID de compte dans l'Amazon Resource Name (ARN) pour les documents publics AWS (documents commençant par `AWS-*`).

Exemples

- Exemple 1 : Autoriser un utilisateur à effectuer des opérations Systems Manager dans une seule région
- Exemple 2 : Autoriser un utilisateur à répertorier les documents pour une seule région

Exemple 1 : Autoriser un utilisateur à effectuer des opérations Systems Manager dans une seule région

L'exemple suivant accorde des autorisations pour effectuer des opérations Systems Manager uniquement dans la Région USA Est (Ohio) (us-east-2).

```
{
  "Version": "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : [
        "ssm:*"
      ],
      "Resource" : [
        "arn:aws:ssm:us-east-2:aws-account-ID:*"
      ]
    }
  ]
}
```

Exemple 2 : Autoriser un utilisateur à répertorier les documents pour une seule région

L'exemple suivant accorde des permissions pour répertorier tous les noms de documents qui commencent par **`Update`** dans la Région USA Est (Ohio) (us-east-2).

```
{
  "Version": "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : [
        "ssm:ListDocuments"
      ],
      "Resource" : [
        "arn:aws:ssm:us-east-2:aws-account-ID:document/Update*"
      ]
    }
  ]
}
```

Exemple 3 : Autoriser un utilisateur à utiliser un document SSM spécifique pour exécuter des commandes sur des nœuds spécifiques

L'exemple de politique IAM suivant permet à un utilisateur d'effectuer les opérations suivantes dans la Région USA Est (Ohio) (us-east-2) :

- Répertorier les documents Systems Manager (documents SSM) et les versions de documents.
- Afficher les détails des documents.
- Envoyer une commande à l'aide du document indiqué dans la politique. Le nom du document est défini par l'entrée suivante.

  ```
  arn:aws:ssm:us-east-2:aws-account-ID:document/Systems-Manager-document-name
  ```

- Envoyer une commande à trois nœuds. Les nœuds sont déterminés 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 des flux de travail dans Automation, une fonctionnalité de AWS Systems Manager.
- Obtenir des informations sur les flux de travail Automation.

Si vous souhaitez accorder à un utilisateur l'autorisation d'utiliser ce document pour envoyer des commandes sur n'importe quel nœud auquel l'utilisateur a accès, vous pouvez spécifier l'entrée suivante dans la section `Resource` et supprimer les autres entrées de nœud. L'exemple utilise la Région USA Est (Ohio) (us-east-2).

```
"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:us-east-2:aws-account-ID:automation-definition/*"
            ]
        },
        {
            "Action": "ssm:DescribeAutomationExecutions",
            "Effect": "Allow",
            "Resource": [
                "*"
            ]
        },
        {
            "Action": [
                "ssm:StopAutomationExecution",
                "ssm:GetAutomationExecution"
            ],
            "Effect": "Allow",
            "Resource": [
                "*"
            ]
        }
    ]
}
```

Affichage de documents Systems Manager basés sur des balises

Vous pouvez utiliser des conditions dans votre politique basée sur l'identité pour contrôler l'accès aux ressources Systems Manager en fonction des balises. Cet exemple montre comment créer une

politique 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 politique 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 de votre compte. Si un utilisateur nommé `richard-roe` tente d'afficher un document Systems Manager, le document doit avoir la balise `Owner=richard-roe` ou `owner=richard-roe`. Dans le cas contraire, l'accès est refusé. 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 Conditions pour éléments de politique JSON IAM dans le Guide de l'utilisateur IAM.

AWS politiques gérées pour AWS Systems Manager

Une politique AWS gérée est une politique autonome créée et administrée par AWS. AWS les politiques gérées sont conçues pour fournir des autorisations pour de nombreux cas d'utilisation courants afin que vous puissiez commencer à attribuer des autorisations aux utilisateurs, aux groupes et aux rôles.

N'oubliez pas que les politiques AWS gérées peuvent ne pas accorder d'autorisations de moindre privilège pour vos cas d'utilisation spécifiques, car elles sont accessibles à tous les AWS clients. Nous vous recommandons de réduire encore les autorisations en définissant des politiques gérées par le client qui sont propres à vos cas d'utilisation.

Vous ne pouvez pas modifier les autorisations définies dans les politiques AWS gérées. Si les autorisations définies dans une politique AWS gérée sont AWS mises à jour, la mise à jour affecte toutes les identités principales (utilisateurs, groupes et rôles) auxquelles la politique est attachée. AWS est le plus susceptible de mettre à jour une politique AWS gérée lorsqu'une nouvelle Service AWS est lancée ou lorsque de nouvelles opérations d'API sont disponibles pour les services existants.

Pour plus d'informations, consultez la section Politiques gérées par AWS dans le Guide de l'utilisateur IAM.

AWS politique gérée : Politique AmazonSSM ServiceRole

Vous ne pouvez pas vous associer `AmazonSSMServiceRolePolicy` à vos entités AWS Identity and Access Management (IAM). Cette politique est associée à un rôle lié à un service qui permet d' AWS Systems Manager effectuer des actions en votre nom. Pour plus d'informations, consultez Utilisation des rôles pour collecter l'inventaire et consulter OpsData.

Sauf mention contraire, `AmazonSSMServiceRolePolicy` 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:StopAutomationExecution`

- ssm:ListTagsForResource
- ssm:GetCalendarState
- ssm:UpdateServiceSetting [1]
- ssm:GetServiceSetting [1]
- ec2:DescribeInstanceAttribute
- ec2:DescribeInstanceStatus
- ec2:DescribeInstances
- lambda:InvokeFunction [2]
- states:DescribeExecution [3]
- states:StartExecution [3]
- resource-groups:ListGroups
- resource-groups:ListGroupResources
- resource-groups:GetGroupQuery
- tag:GetResources
- config:SelectResourceConfig
- config:DescribeComplianceByConfigRule
- config:DescribeComplianceByResource
- config:DescribeRemediationConfigurations
- config:DescribeConfigurationRecorders
- cloudwatch:DescribeAlarms
- compute-optimizer:GetEC2InstanceRecommendations
- compute-optimizer:GetEnrollmentStatus
- support:DescribeTrustedAdvisorChecks
- support:DescribeTrustedAdvisorCheckSummaries
- support:DescribeTrustedAdvisorCheckResult
- support:DescribeCases
- iam:PassRole [4]
- cloudformation:DescribeStacks
- cloudformation:ListStackResources
- cloudformation:ListStackInstances [5]

- `cloudformation:DescribeStackSetOperation` [5]
- `cloudformation:DeleteStackSet` [5]
- `cloudformation:DeleteStackInstances` [6]
- `events:PutRule` [7]
- `events:PutTargets` [7]
- `events:RemoveTargets` [8]
- `events:DeleteRule` [8]
- `events:DescribeRule`
- `securityhub:DescribeHub`

[1] Les actions `ssm:UpdateServiceSetting` et `ssm:GetServiceSetting` ne sont autorisées que pour les ressources suivantes.

```
arn:aws:ssm:*:*:servicesetting/ssm/opsitem/*
arn:aws:ssm:*:*:servicesetting/ssm/opsdata/*
```

[2] L'action `lambda:InvokeFunction` n'est autorisée que pour les ressources suivantes.

```
arn:aws:lambda:*:*:function:SSM*
arn:aws:lambda:*:*:function:*:SSM*
```

[3] Les actions `states:` ne sont autorisées que sur les ressources suivantes.

```
arn:aws:states:*:*:stateMachine:SSM*
arn:aws:states:*:*:execution:SSM*
```

[4] L'action `iam:PassRole` n'est autorisée par la condition suivante que pour le service Systems Manager.

```
"Condition": {
   "StringEquals": {
      "iam:PassedToService": [
         "ssm.amazonaws.com"
      ]
   }
}
```

[5] Les actions `cloudformation:ListStackInstances`, `cloudformation:DescribeStackSetOperation` et `cloudformation:DeleteStackSet` sont autorisées sur la ressource suivante uniquement.

```
arn:aws:cloudformation:*:*:stackset/AWS-QuickSetup-SSM*:*
```

[6] L'action `cloudformation:DeleteStackInstances` n'est autorisée que pour les ressources suivantes.

```
arn:aws:cloudformation:*:*:stackset/AWS-QuickSetup-SSM*:*
arn:aws:cloudformation:*:*:stackset-target/AWS-QuickSetup-SSM*:*
arn:aws:cloudformation:*:*:type/resource/*
```

[7] Les actions `events:PutRule` et `events:PutTargets` ne sont autorisées par la condition suivante que pour le service Systems Manager.

```
"Condition": {
    "StringEquals": {
        "events:ManagedBy": "ssm.amazonaws.com"
    }
}
```

[8] Les actions `events:RemoveTargets` et `events:DeleteRule` ne sont autorisées que sur la ressource suivante.

```
arn:aws:events:*:*:rule/SSMExplorerManagedRule
```

Pour en savoir plus sur la politique, y compris la dernière version du document de politique JSON, consultez la politique d'AmazonSSM dans le ServiceRole Guide de référence des politiques AWS gérées.

AWS politique gérée : AmazonSSM Access ReadOnly

Vous pouvez associer la politique `AmazonSSMReadOnlyAccess` à vos identités IAM. Cette politique accorde un accès en lecture seule aux opérations d' AWS Systems Manager API`Describe*`, notamment`Get*`, et. `List*`

Pour en savoir plus sur la politique, y compris la dernière version du document de politique JSON, consultez AmazonSSM ReadOnly Access dans le Guide de référence des politiques AWS gérées.

AWS politique gérée : AWSSystemsManagerOpsDataSyncServiceRolePolicy

Vous ne pouvez pas joindre de `AWSSystemsManagerOpsDataSyncServiceRolePolicy` à vos entités IAM. Cette politique est attachée à un rôle lié au service qui permet à Systems Manager d'effectuer des actions en votre nom. Pour plus d'informations, consultez Utiliser des rôles pour créer OpsData et OpsItems pour Explorer.

`AWSSystemsManagerOpsDataSyncServiceRolePolicy`permet au rôle `AWSServiceRoleForSystemsManagerOpsDataSync` lié au service de créer et de mettre à jour OpsItems et à OpsData partir AWS Security Hub des résultats.

Sauf mention contraire, la politique permet à Systems Manager d'effectuer les actions suivantes sur toutes les ressources connexes (`"Resource": "*"`) :

- `ssm:GetOpsItem` [1]
- `ssm:UpdateOpsItem` [1]
- `ssm:CreateOpsItem`
- `ssm:AddTagsToResource` [2]
- `ssm:UpdateServiceSetting` [3]
- `ssm:GetServiceSetting` [3]
- `securityhub:GetFindings`
- `securityhub:GetFindings`
- `securityhub:BatchUpdateFindings` [4]

[1] Les actions `ssm:GetOpsItem` et `ssm:UpdateOpsItem` ne sont autorisées par la condition suivante que pour le service Systems Manager.

```
"Condition": {
    "StringEquals": {
        "aws:ResourceTag/ExplorerSecurityHubOpsItem": "true"
    }
}
```

[2] L'action `ssm:AddTagsToResource` n'est autorisée que pour la ressource suivante.

```
arn:aws:ssm:*:*:opsitem/*
```

[3] Les actions `ssm:UpdateServiceSetting` et `ssm:GetServiceSetting` ne sont autorisées que pour les ressources suivantes.

```
arn:aws:ssm:*:*:servicesetting/ssm/opsitem/*
arn:aws:ssm:*:*:servicesetting/ssm/opsdata/*
```

[4] Les autorisations sont refusées aux `securityhub:BatchUpdateFindings` par la condition suivante, pour le service Systems Manager uniquement.

```
{
   "Effect": "Deny",
   "Action": "securityhub:BatchUpdateFindings",
   "Resource": "*",
   "Condition": {
    "StringEquals": {
     "securityhub:ASFFSyntaxPath/Workflow.Status": "SUPPRESSED"
    }
   }
  },
  {
   "Effect": "Deny",
   "Action": "securityhub:BatchUpdateFindings",
   "Resource": "*",
   "Condition": {
    "Null": {
     "securityhub:ASFFSyntaxPath/Confidence": false
    }
   }
  },
  {
   "Effect": "Deny",
   "Action": "securityhub:BatchUpdateFindings",
   "Resource": "*",
   "Condition": {
    "Null": {
     "securityhub:ASFFSyntaxPath/Criticality": false
    }
   }
  },
  {
   "Effect": "Deny",
   "Action": "securityhub:BatchUpdateFindings",
   "Resource": "*",
```

```
    "Condition": {
     "Null": {
      "securityhub:ASFFSyntaxPath/Note.Text": false
     }
    }
   },
   {
    "Effect": "Deny",
    "Action": "securityhub:BatchUpdateFindings",
    "Resource": "*",
    "Condition": {
     "Null": {
      "securityhub:ASFFSyntaxPath/Note.UpdatedBy": false
     }
    }
   },
   {
    "Effect": "Deny",
    "Action": "securityhub:BatchUpdateFindings",
    "Resource": "*",
    "Condition": {
     "Null": {
      "securityhub:ASFFSyntaxPath/RelatedFindings": false
     }
    }
   },
   {
    "Effect": "Deny",
    "Action": "securityhub:BatchUpdateFindings",
    "Resource": "*",
    "Condition": {
     "Null": {
      "securityhub:ASFFSyntaxPath/Types": false
     }
    }
   },
   {
    "Effect": "Deny",
    "Action": "securityhub:BatchUpdateFindings",
    "Resource": "*",
    "Condition": {
     "Null": {
      "securityhub:ASFFSyntaxPath/UserDefinedFields.key": false
     }
```

```
    }
   },
   {
    "Effect": "Deny",
    "Action": "securityhub:BatchUpdateFindings",
    "Resource": "*",
    "Condition": {
     "Null": {
      "securityhub:ASFFSyntaxPath/UserDefinedFields.value": false
     }
    }
   },
   {
    "Effect": "Deny",
    "Action": "securityhub:BatchUpdateFindings",
    "Resource": "*",
    "Condition": {
     "Null": {
      "securityhub:ASFFSyntaxPath/VerificationState": false
     }
    }
```

Pour plus de détails sur la politique, y compris la dernière version du document de politique JSON, consultez AWSSystemsManagerOpsDataSyncServiceRolePolicyle Guide de référence des politiques AWS gérées.

AWS stratégie gérée : politique InstanceDefault AmazonSSMManageDec2

Vous ne devez associer `AmazonSSMManagedEC2InstanceDefaultPolicy` aux rôles IAM que les instances Amazon EC2 pour lesquelles vous souhaitez être autorisé à Systems Manager utiliser les fonctionnalités. Vous ne devez pas associer ce rôle à d'autres entités IAM, telles que les utilisateurs IAM et les groupes IAM, ni à des rôles IAM ayant d'autres objectifs. Pour plus d'informations, consultez Utilisation du paramètre de configuration de gestion d'hôte par défaut.

Cette politique octroie des autorisations permettant au SSM Agent de votre instance Amazon EC2 de récupérer des documents, exécutez des commandes à l'aide de Run Command, établir des sessions à l'aide de Session Manager, collecter un inventaire de l'instance et rechercher les correctifs et leur conformité à l'aide de Patch Manager.

Systems Manager utilise un jeton d'autorisation personnalisé pour chaque instance afin de garantir l'exécution des opérations d'API par SSM Agent sur la bonne instance. Systems Manager valide le

jeton d'autorisation personnalisé par rapport à l'Amazon Resource Name (ARN) de l'instance, fourni lors de l'opération d'API.

La `AmazonSSMManagedEC2InstanceDefaultPolicy` stratégie d'autorisations d'un rôle Systems Manager permet d'effectuer les actions suivantes sur toutes les ressources associées:

- `ssm:DescribeAssociation`
- `ssm:GetDeployablePatchSnapshotForInstance`
- `ssm:GetDocument`
- `ssm:DescribeDocument`
- `ssm:GetManifest`
- `ssm:ListAssociations`
- `ssm:ListInstanceAssociations`
- `ssm:PutInventory`
- `ssm:PutComplianceItems`
- `ssm:PutConfigurePackageResult`
- `ssm:UpdateAssociationStatus`
- `ssm:UpdateInstanceAssociationStatus`
- `ssm:UpdateInstanceInformation`
- `ssmmessages:CreateControlChannel`
- `ssmmessages:CreateDataChannel`
- `ssmmessages:OpenControlChannel`
- `ssmmessages:OpenDataChannel`
- `ec2messages:AcknowledgeMessage`
- `ec2messages:DeleteMessage`
- `ec2messages:FailMessage`
- `ec2messages:GetEndpoint`
- `ec2messages:GetMessages`
- `ec2messages:SendReply`

Pour plus de détails sur la politique, y compris la dernière version du document de politique JSON, consultez la politique AmazonSSMManageDec2 dans le Guide de référence des InstanceDefault politiques gérées.AWS

Systems Managermises à jour des politiques AWS gérées

Dans le tableau suivant, consultez les détails des mises à jour des politiques AWS gérées Systems Manager depuis que ce service a commencé à suivre ces modifications le 12 mars 2021. Pour plus d'informations sur les autres politiques gérées pour le service Systems Manager, voir Politiques gérées supplémentaires pour Systems Manager plus loin dans cette rubrique. Pour obtenir des alertes automatiques sur les modifications apportées à cette page, abonnez-vous au flux RSS de la page Systems Manager Historique du document.

Modification	Description	Date
AWSSystemsManagerOpsDataSyncServiceRolePolicy - Mettre à jour vers une politique existante.	OpsCentera mis à jour la politique afin d'améliorer la sécurité du code de service dans le cadre du rôle lié au service Explorer afin de gérer les opérations OpsData connexes.	28 juin 2023
AmazonSSMManagedEC2InstanceDefaultPolicy : nouvelle politique.	Systems Manager a ajouté une nouvelle politique pour autoriser la fonctionnalité Systems Manager sur les instances Amazon EC2 sans utiliser de profil d'instance IAM.	18 août 2022
ServiceRolePolitique Amazon SSM : mise à jour d'une politique existante.	Systems Manager a ajouté de nouvelles autorisations pour autoriser Explorer à créer une règle gérée lorsque vous activez Security Hub depuis	27 avril 2021

Modification	Description	Date
	Explorer ou OpsCenter. De nouvelles autorisations ont été ajoutées pour vérifier que la configuration et l'optimis eur de calcul répondent aux exigences nécessaires avant d'autoriser. OpsData	
AWSSystemsManagerOpsDataSyncServiceRolePolicy : nouvelle politique.	Systems Managera ajouté une nouvelle politique pour créer et mettre à jour OpsItems et à OpsData partir des conclusio ns du Security Hub dans Explorer etOpsCenter.	27 avril 2021
AmazonSSMServiceRolePolicy - Mettre à jour vers une politique existante.	Systems Managera ajouté de nouvelles autorisations pour permettre l'affichage des agrégats OpsData et des OpsItems détails provenant de plusieurs comptes et Régions AWS dansExplorer.	24 mars 2021
Systems Manager a démarré le suivi des modifications	Systems Managera commencé à suivre les modifications apportées AWS à ses politiques gérées.	12 mars 2021

Politiques gérées supplémentaires pour Systems Manager

Outre les politiques gérées décrites précédemment dans cette rubrique, les politiques suivantes sont également prises en charge par Systems Manager.

- AmazonSSMAutomationApproverAccess: politique AWS gérée qui permet d'accéder à la visualisation des exécutions automatisées et d'envoyer les décisions d'approbation à l'automatisation en attente d'approbation.

- AmazonSSMAutomationRole— politique AWS gérée qui autorise le service Systems Manager Automation à exécuter les activités définies dans les runbooks Automation. Attribuez cette politique aux administrateurs et aux utilisateurs avancés de confiance.
- AmazonSSMDirectoryServiceAccess— politique AWS gérée qui permet d'SSM Agentaccéder au AWS Directory Service nom de l'utilisateur aux demandes d'adhésion au domaine par le nœud géré.
- AmazonSSMFullAccess— politique AWS gérée qui accorde un accès complet à l'Systems ManagerAPI et aux documents.
- AmazonSSMMaintenanceWindowRole— politique AWS gérée qui fournit aux fenêtres de maintenance des autorisations d'accès à l'API Systems Manager.
- AmazonSSMManagedInstanceCore : politique gérée par AWS permettant à un nœud d'utiliser une fonctionnalité principale de service de Systems Manager.
- AmazonSSMPatchAssociation: politique AWS gérée qui permet d'accéder aux instances enfants pour les opérations d'association de correctifs.
- AmazonSSMReadOnlyAccess— politique AWS gérée qui accorde l'accès aux opérations d'API en Systems Manager lecture seule, telles que `Get*` et. `List*`
- AWSSSMOpsInsightsServiceRolePolicy: politique AWS gérée qui fournit des autorisations pour créer et mettre à jour des informations opérationnelles OpsItemsdansSystems Manager. Utilisé pour fournir des autorisations via le rôle lié au service. AWSServiceRoleForAmazonSSM_OpsInsights
- AWSSystemsManagerAccountDiscoveryServicePolicy— politique AWS gérée qui autorise Systems Manager à découvrir Compte AWS des informations.
- AWSSystemsManagerChangeManagementServicePolicy: politique AWS gérée qui donne accès aux AWS ressources gérées ou utilisées par le cadre de gestion des Systems Manager modifications et utilisées par le rôle lié au service. `AWSServiceRoleForSystemsManagerChangeManagement`
- AmazonEC2RoleforSSM— Cette politique n'est plus prise en charge et ne doit pas être utilisée. À la place, utilisez la `AmazonSSMManagedInstanceCore` politique pour autoriser les fonctionnalités de base du Systems Manager service sur les instances EC2. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager.

Résolution des problèmes d'identité et d'accès avec AWS Systems Manager

Utilisez les informations suivantes pour identifier et résoudre les problèmes courants que vous pouvez rencontrer lorsque vous travaillez avec AWS Systems Manager et AWS Identity and Access Management (IAM).

Rubriques

Je ne suis pas autorisé à effectuer une action dans Systems Manager

Si la 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 vos informations de connexion.

L'exemple d'erreur suivant se produit lorsque l'utilisateur IAM `mateojackson` tente d'utiliser la console pour afficher des informations détaillées concernant un document, mais ne dispose pas des autorisations `ssm:GetDocument` nécessaires.

```
User: arn:aws:ssm::123456789012:user/mateojackson isn't authorized to perform:
 ssm:GetDocument on resource: MyExampleDocument
```

Dans ce cas, Mateo demande à son administrateur de mettre à jour ses politiques pour lui permettre d'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 une erreur selon laquelle vous n'êtes pas autorisé à exécuter `iam:PassRole` l'action, vos stratégies doivent être mises à jour afin de 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, les stratégies de Mary doivent être mises à jour pour lui permettre d'exécuter l'action `iam:PassRole`.

Si vous avez encore besoin d'aide, contactez votre administrateur AWS. Votre administrateur vous a fourni vos informations de connexion.

Je veux autoriser des personnes extérieures à mon Compte AWS à 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 pourront utiliser pour accéder à vos ressources. Vous pouvez spécifier la personne à qui vous souhaitez confier le rôle. Pour les services qui prennent en charge les politiques basées sur les ressources ou les listes de contrôle d'accès (ACL), vous pouvez utiliser ces politiques pour donner 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.
- Pour savoir comment octroyer l'accès à vos ressources à des Comptes AWS dont vous êtes propriétaire, consultez la section Fournir l'accès à un utilisateur IAM dans un autre Compte AWS que vous possédez dans le Guide de l'utilisateur IAM.
- Pour savoir comment octroyer l'accès à vos ressources à des Comptes AWS tiers, consultez Fournir l'accès aux Comptes AWS appartenant à des tiers dans le Guide de l'utilisateur IAM.
- Pour savoir comment fournir un accès par le biais de la fédération d'identité, consultez Fournir un accès à des utilisateurs authentifiés en externe (fédération d'identité) dans le Guide de l'utilisateur IAM.
- Pour découvrir quelle est la différence entre l'utilisation des rôles et l'utilisation des politiques basées sur les ressources pour l'accès entre comptes, consultez Différence entre les rôles IAM et les politiques basées sur les ressources dans le Guide de l'utilisateur IAM.

Utilisation des rôles liés aux services pour Systems Manager

AWS Systems Managerutilise des AWS Identity and Access Management rôles liés à un service (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 incluent toutes les autorisations requises par le service pour appeler d'autres Services AWS en votre nom.

> **Note**
>
> Une fonction de service est différente d'un rôle lié à un service. Un rôle de service est un type de rôle AWS Identity and Access Management (IAM) qui accorde des autorisations à un service Service AWS afin qu'il puisse accéder aux AWS ressources. Seuls quelques scénarios Systems Manager nécessitent un rôle de service. Lorsque vous créez une fonction du service pour Systems Manager, vous choisissez les autorisations à accorder pour qu'il puisse accéder aux autres ressources AWS ou interagir avec ces dernières.

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 politique d'approbation et la politique d'autorisation. De plus, cette politique d'autorisation ne peut pas être attachée à une autre entité IAM.

Vous pouvez supprimer un rôle lié à un service uniquement après la suppression préalable de ses ressources connexes. Vos ressources Systems Manager sont ainsi protégées, car vous ne pouvez pas involontairement supprimer l'autorisation d'accéder aux ressources.

> **Note**
>
> Pour les nœuds non EC2 dans un environnement hybride et multicloud, vous avez besoin d'un rôle IAM supplémentaire qui permet à ces machines de communiquer avec le service Systems Manager. Il s'agit du rôle de service IAM pour Systems Manager. Ce rôle accorde AWS Security Token Service (AWS STS) AssumeRoleconfiance au Systems Manager service. 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 à AWS des ressources auxquelles vous n'avez pas normalement accès. Pour plus d'informations, consultez la section Créer le rôle de service IAM requis pour Systems Manager dans les

environnements hybrides et multicloud et AssumeRoledans le Guide de référence des AWS Security Token Service API.

Pour plus d'informations sur les autres services prenant en charge les rôles liés à un service, reportez-vous aux Services AWS opérationnels avec IAM et recherchez les services présentant la mention Yes (Oui) dans la colonne Service-linked roles (Rôles liés à un service). Choisissez un Oui ayant un lien permettant de consulter les détails du rôle pour ce service.

Rubriques

- Utilisation des rôles pour collecter l'inventaire et consulter OpsData
- Utilisation des rôles pour collecter des Compte AWS informations pour OpsCenter et Explorer
- Utiliser des rôles pour créer OpsData et OpsItems pour Explorer
- L'utilisation des rôles pour la création d'informations opérationnelles d'OpsItems dans l'OpsCenter de Systems Manager
- Utiliser des rôles pour exporter Explorer OpsData

Utilisation des rôles pour collecter l'inventaire et consulter OpsData

Systems Managerutilise le rôle lié au service nommé. **`AWSServiceRoleForAmazonSSM`** AWS Systems Manager utilise ce rôle de service IAM pour gérer les AWS ressources en votre nom.

Autorisations de rôle liées au service pour l'inventaire, et OpsData OpsItems

Le rôle lié à un service `AWSServiceRoleForAmazonSSM` fait confiance uniquement à `ssm.amazonaws.com` pour assumer le rôle.

Vous pouvez utiliser le rôle lié au service `AWSServiceRoleForAmazonSSM` de Systems Manager dans les cas suivants :

- La fonctionnalité d'inventaire Systems Manager utilise le rôle lié à un service `AWSServiceRoleForAmazonSSM` pour collecter les métadonnées d'inventaire à partir des balises et des groupes de ressources.
- La Explorer fonctionnalité utilise le rôle lié au service `AWSServiceRoleForAmazonSSM` pour permettre l'affichage OpsData et OpsItems depuis plusieurs comptes. Ce rôle lié au service autorise également Explorer à créer une règle gérée lorsque vous autorisez Security Hub en tant que source de données depuis Explorer ou OpsCenter.

> **Important**
>
> Auparavant, la console Systems Manager vous permettait de choisir le rôle lié au service IAM AWS géré `AWSServiceRoleForAmazonSSM` à utiliser comme rôle de maintenance pour vos tâches. L'utilisation de ce rôle et de la politique associée, `AmazonSSMServiceRolePolicy`, pour les tâches de la fenêtre de maintenance n'est plus recommandée. Si vous utilisez ce rôle pour des tâches de fenêtre de maintenance maintenant, nous vous encourageons à cesser de l'utiliser. Au lieu de cela, créez votre propre rôle IAM permettant la communication entre Systems Manager et d'autres Services AWS lorsque les tâches de votre fenêtre de maintenance sont exécutées.
> Pour plus d'informations, consultez Configuration de Maintenance Windows.

La politique gérée utilisée pour fournir des autorisations au rôle `AWSServiceRoleForAmazonSSM` est `AmazonSSMServiceRolePolicy`. Pour plus d'informations sur les autorisations accordées, consultez AWS politique gérée : Politique AmazonSSM ServiceRole.

Création du rôle lié au service **`AWSServiceRoleForAmazonSSM`** pour Systems Manager

Vous pouvez utiliser la console IAM pour créer un rôle lié au service avec le cas d'utilisation EC2. L'utilisation de commandes pour IAM dans la AWS Command Line Interface (AWS CLI) ou l'utilisation de l'API IAM, crée un rôle lié au service qui porte le nom du service `ssm.amazonaws.com`. Pour plus d'informations, consultez Création d'un rôle lié à un service dans le Guide de l'utilisateur IAM.

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 du rôle lié au service **`AWSServiceRoleForAmazonSSM`** 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 changer le nom du rôle, car plusieurs entités peuvent faire référence au rôle. Néanmoins, vous pouvez modifier la description du rôle à l'aide d'IAM. Pour en savoir plus, consultez Modification d'un rôle lié à un service dans le guide de l'utilisateur IAM.

Suppression du rôle lié au service **AWSServiceRoleForAmazonSSM** 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, aucune entité inutilisée n'est surveillée ou gérée activement. Vous pouvez utiliser la console IAM AWS CLI, 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 au service `AWSServiceRoleForAmazonSSM` peut être utilisé par plusieurs fonctionnalités, 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 la capacité Inventaire, les données d'inventaire pour les balises et les groupes de ressources ne seront plus synchronisées. Vous devez nettoyer les ressources de votre rôle lié à un service avant de pouvoir les supprimer manuellement.
- Explorer: Si vous supprimez le rôle lié au service utilisé par la Explorer fonctionnalité, les rôles entre comptes et entre régions OpsItems ne sont plus OpsData visibles.

> **Note**
>
> Si le service Systems Manager utilise le rôle lorsque vous essayez de supprimer les balises ou les groupes de ressources, la suppression peut échouer. Si cela se produit, patientez quelques minutes et réessayez.

Pour supprimer les ressources Systems Manager utilisées par le service **AWSServiceRoleForAmazonSSM**

1. Pour supprimer des balises, consultez Ajout et suppression de balises sur une ressource individuelle.
2. Pour supprimer des groupes de ressources, voir Supprimer des groupes de AWS Resource Groups.

Pour supprimer manuellement le rôle lié au service **AWSServiceRoleForAmazonSSM** à l'aide d'IAM

Utilisez la console IAM AWS CLI, ou l'API IAM pour supprimer le rôle lié au `AWSServiceRoleForAmazonSSM` service. Pour plus d'informations, consultez Suppression d'un rôle lié à un service dans le Guide de l'utilisateur IAM.

Régions prises en charge pour le rôle lié à un service **`AWSServiceRoleForAmazonSSM`** de Systems Manager

Systems Managerprend en charge l'utilisation du rôle `AWSServiceRoleForAmazonSSM` lié au service partout Régions AWS où le service est disponible. Pour plus d'informations, consultez Points de terminaison et quotas AWS Systems Manager.

Utilisation des rôles pour collecter des Compte AWS informations pour OpsCenter et Explorer

Systems Managerutilise le rôle lié au service nommé. **`AWSServiceRoleForAmazonSSM_AccountDiscovery`** AWS Systems Manager utilise ce rôle de service IAM pour appeler d'autres personnes afin Services AWS de découvrir des Compte AWS informations.

Autorisations des rôles liés à un service pour la découverte de comptes Systems Manager

Le rôle lié à un service `AWSServiceRoleForAmazonSSM_AccountDiscovery` approuve les services suivants pour endosser le rôle :

- `accountdiscovery.ssm.amazonaws.com`

La politique d'autorisations liée au rôle permet à Systems Manager de réaliser les actions suivantes sur les ressources spécifiées :

- `organizations:DescribeAccount`
- `organizations:DescribeOrganizationalUnit`
- `organizations:DescribeOrganization`
- `organizations:ListAccounts`
- `organizations:ListAWSServiceAccessForOrganization`
- `organizations:ListChildren`
- `organizations:ListParents`

- `organizations:ListDelegatedServicesForAccount`
- `organizations:ListDelegatedAdministrators`
- `organizations:ListRoots`

Vous devez configurer les autorisations de manière à permettre à une entité IAM (comme un utilisateur, un groupe ou un rôle) de créer, modifier ou supprimer un rôle lié à un service. Pour plus d'informations, consultez Autorisations de rôles liés à un service dans le Guide de l'utilisateur IAM.

Création du rôle lié au service **`AWSServiceRoleForAmazonSSM_AccountDiscovery`** pour Systems Manager

Vous devez créer un rôle lié à un service si vous souhaitez utiliser Explorer et OpsCenter, des fonctionnalités de Systems Manager, sur plusieurs Comptes AWS. Pour OpsCenter, vous devez créer manuellement le rôle lié à un service. Pour de plus amples informations, veuillez consulter (Facultatif) Configuration d'OpsCenter pour gérer de manière centralisée les OpsItems de l'ensemble des comptes.

Pour Explorer, si vous créez une synchronisation des données de ressource avec Systems Manager dans AWS Management Console, vous pouvez créer le rôle lié à un service en choisissant le bouton Create role (Créer un rôle). Pour créer une synchronisation des données de ressource par programmation, vous devez créer le rôle au préalable. Vous pouvez créer le rôle à l'aide de l'opération CreateServiceLinkedRoleAPI.

Modification du rôle lié au service **`AWSServiceRoleForAmazonSSM_AccountDiscovery`** 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 changer le nom du rôle, car plusieurs entités peuvent faire référence au rôle. Néanmoins, vous pouvez modifier la description du rôle à l'aide d'IAM. Pour en savoir plus, consultez Modification d'un rôle lié à un service dans le guide de l'utilisateur IAM.

Suppression du rôle lié au service **`AWSServiceRoleForAmazonSSM_AccountDiscovery`** 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, aucune entité inutilisée n'est

surveillée ou gérée activement. Cependant, vous devez nettoyer votre rôle lié à un service avant de pouvoir le supprimer manuellement.

Nettoyage du rôle lié au service **`AWSServiceRoleForAmazonSSM_AccountDiscovery`**

Avant de pouvoir utiliser IAM pour supprimer le rôle lié au service `AWSServiceRoleForAmazonSSM_AccountDiscovery`, vous devez d'abord supprimer toutes les synchronisations de données des ressources Explorer. Pour de plus amples informations, veuillez consulter Suppression des données de ressource Systems Manager Explorer.

> **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 au service **`AWSServiceRoleForAmazonSSM_AccountDiscovery`**

Utilisez la console IAM, le AWS CLI, ou l' AWS API pour supprimer le rôle lié au `AWSServiceRoleForAmazonSSM_AccountDiscovery` service. Pour plus d'informations, consultez Suppression d'un rôle lié à un service dans le Guide de l'utilisateur IAM.

Régions prises en charge pour le rôle lié à un service **`AWSServiceRoleForAmazonSSM_AccountDiscovery`** de 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 Points de terminaison et quotas AWS Systems Manager.

Mises à jour du rôle lié à un service AWSServiceRoleForAmazonSSM_AccountDiscovery

Consultez les détails des mises à jour apportées au rôle AWSServiceRoleForAmazonSSM_AccountDiscovery lié au service depuis que ce service a commencé à suivre ces modifications. Pour obtenir des alertes automatiques sur les modifications apportées à cette page, abonnez-vous au flux RSS de la page Systems Manager Historique du document.

Modification	Description	Date
Ajout de nouvelles autorisations	Ce rôle lié à un service comprend désormais les autorisations `organizations:DescribeOrganizationalUnit` et `organizations:ListRoots`. Ces autorisations permettent à un compte AWS Organizations de gestion ou à un compte administrateur délégué de Systems Manager de travailler avec OpsItems plusieurs comptes. Pour de plus amples informations, veuillez consulter (Facultatif) Configuration d'OpsCenter pour gérer de manière centralisée les OpsItems de l'ensemble des comptes.	17 octobre 2022

Utiliser des rôles pour créer OpsData et OpsItems pour Explorer

Systems Managerutilise le rôle lié au service nommé. **`AWSServiceRoleForSystemsManagerOpsDataSync`** AWS Systems Manager utilise ce rôle de service IAM pour Explorer créer OpsData etOpsItems.

Autorisations de rôle liées au service pour la synchronisation Systems Manager OpsData

Le rôle lié à un service `AWSServiceRoleForSystemsManagerOpsDataSync` approuve les services suivants pour endosser le rôle :

- `opsdatasync.ssm.amazonaws.com`

La politique d'autorisations liée au rôle permet à Systems Manager de réaliser les actions suivantes sur les ressources spécifiées :

- Systems Manager Explorer exige qu'un rôle lié à un service octroie l'autorisation de mettre à jour un résultat lié à la sécurité lorsqu'un OpsItem est mis à jour, de créer et mettre à jour un OpsItem, et de désactiver la source de données Security Hub lorsqu'une règle gérée SSM est supprimée par les clients.

La politique gérée utilisée pour fournir des autorisations au rôle `AWSServiceRoleForSystemsManagerOpsDataSync` est `AWSSystemsManagerOpsDataSyncServiceRolePolicy`. Pour plus d'informations sur les autorisations accordées, consultez AWS politique gérée : AWSSystemsManagerOpsDataSyncServiceRolePolicy.

Vous devez configurer les autorisations de manière à permettre à une entité IAM (comme un utilisateur, un groupe ou un rôle) de créer, modifier ou supprimer un rôle lié à un service. Pour plus d'informations, consultez Autorisations de rôles liés à un service dans le Guide de l'utilisateur IAM.

Création du rôle lié au service **`AWSServiceRoleForSystemsManagerOpsDataSync`** pour Systems Manager

Vous n'avez pas besoin de créer manuellement un rôle lié à un service. Lorsque vous l'activez Explorer dans le AWS Management Console, Systems Manager crée le rôle lié au service pour vous.

> **Important**
>
> Ce rôle lié au service peut apparaître dans votre compte si vous avez effectué une action dans un autre service qui utilise les fonctions prises en charge par ce rôle. De même, si vous utilisiez Systems Manager avant le 1er janvier 2017, quand il commençait à prendre en charge les rôles liés à un service, Systems Manager créait le rôle `AWSServiceRoleForSystemsManagerOpsDataSync` dans votre compte. Pour en savoir plus, consultez Un nouveau rôle est apparu dans mon compte IAM.

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 l'activez Explorer dans le AWS Management Console, Systems Manager crée à nouveau le rôle lié au service pour vous.

Vous pouvez également utiliser la console IAM pour créer un rôle lié à un service avec le rôle de AWS service qui permet de créer OpsData et OpsItems d'Explorerutiliser un cas. Dans l'API AWS CLI ou dans l' AWS API, créez un rôle lié à un service avec le nom du `opsdatasync.ssm.amazonaws.com` service. Pour plus d'informations, consultez Création d'un rôle lié à un service dans le Guide de l'utilisateur IAM. Si vous supprimez ce rôle lié à un service, vous pouvez utiliser ce même processus pour créer le rôle à nouveau.

Modification du rôle lié au service **`AWSServiceRoleForSystemsManagerOpsDataSync`** pour Systems Manager

Systems Manager ne vous permet pas de modifier le rôle lié à un service `AWSServiceRoleForSystemsManagerOpsDataSync`. Une fois que vous avez créé un rôle lié à un service, vous ne pouvez pas changer le nom du rôle, car plusieurs entités peuvent faire référence au rôle. Néanmoins, vous pouvez modifier la description du rôle à l'aide d'IAM. Pour en savoir plus, consultez Modification d'un rôle lié à un service dans le guide de l'utilisateur IAM.

Suppression du rôle lié au service **`AWSServiceRoleForSystemsManagerOpsDataSync`** 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, aucune entité inutilisée n'est surveillée ou gérée activement. Cependant, vous devez nettoyer les ressources de votre rôle lié à un service avant de pouvoir les supprimer manuellement.

> **Note**
>
> Si le service 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.

La procédure de suppression des ressources Systems Manager utilisées par le rôle `AWSServiceRoleForSystemsManagerOpsDataSync` dépend du fait que vous avez, ou non, configuré Explorer ou OpsCenter de sorte à s'intégrer à Security Hub.

Pour supprimer les ressources Systems Manager utilisées par le rôle **`AWSServiceRoleForSystemsManagerOpsDataSync`**

- Pour empêcher Explorer de créer des OpsItems pour des résultats de Security Hub, consultez Comment arrêter l'envoi des résultats.
- Pour empêcher OpsCenter de créer de nouveaux OpsItems pour des résultats de Security Hub, consultez

Pour supprimer manuellement le rôle lié au service **`AWSServiceRoleForSystemsManagerOpsDataSync`** à l'aide d'IAM

Utilisez la console IAM, le AWS CLI, ou l' AWS API pour supprimer le rôle lié au `AWSServiceRoleForSystemsManagerOpsDataSync` service. Pour plus d'informations, consultez Suppression d'un rôle lié à un service dans le Guide de l'utilisateur IAM.

Régions prises en charge pour le rôle lié à un service **`AWSServiceRoleForSystemsManagerOpsDataSync`** de 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 de plus amples informations, consultez Points de terminaison et quotas AWS Systems Manager.

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 `AWSServiceRoleForSystemsManagerOpsDataSync` dans les régions suivantes :

Région AWS nom	Identité de la région	Prise en charge dans Systems Manager
US East (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

Région AWS nom	Identité de la région	Prise en charge dans Systems Manager
Asie-Pacifique (Osaka)	ap-northeast-3	Oui
Asie-Pacifique (Séoul)	ap-northeast-2	Oui
Asie-Pacifique (Singapour)	ap-southeast-1	Oui
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
Europe (Stockholm)	eu-north-1	Oui
Amérique du Sud (São Paulo)	sa-east-1	Oui
AWS GovCloud (US)	us-gov-west-1	Non

L'utilisation des rôles pour la création d'informations opérationnelles d'OpsItems dans l'OpsCenter de Systems Manager

Systems Manager utilise le rôle lié au service appelé **`AWSServiceRoleForAmazonSSM_OpsInsights`**. AWS Systems Manager utilise cette fonction du service IAM pour créer et mettre à jour des informations opérationnelles OpsItems dans Systems Manager OpsCenter.

Autorisations de rôle lié à un service **AWSServiceRoleForAmazonSSM_OpsInsights** pour les OpsItems d'informations opérationnelles de Systems Manager

Le rôle lié à un service `AWSServiceRoleForAmazonSSM_OpsInsights` approuve les services suivants pour endosser le rôle :

- `opsinsights.ssm.amazonaws.com`

La politique d'autorisations liée au rôle permet à Systems Manager de réaliser les actions suivantes sur les ressources spécifiées :

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "AllowCreateOpsItem",
   "Effect": "Allow",
   "Action": [
    "ssm:CreateOpsItem",
    "ssm:AddTagsToResource"
   ],
   "Resource": "*"
  },
  {
   "Sid": "AllowAccessOpsItem",
   "Effect": "Allow",
   "Action": [
    "ssm:UpdateOpsItem",
    "ssm:GetOpsItem"
   ],
   "Resource": "*",
   "Condition": {
    "StringEquals": {
     "aws:ResourceTag/SsmOperationalInsight": "true"
    }
   }
  }
 ]
}
```

Vous devez configurer les autorisations de manière à permettre à une entité IAM (comme un utilisateur, un groupe ou un rôle) de créer, modifier ou supprimer un rôle lié à un service. Pour plus d'informations, consultez Autorisations de rôles liés à un service dans le Guide de l'utilisateur IAM.

Création du rôle lié au service **AWSServiceRoleForAmazonSSM_OpsInsights** pour Systems Manager

Vous devez créer un rôle lié au service. Si vous activez les informations opérationnelles avec Systems Manager dans la AWS Management Console, vous pouvez créer le rôle lié au service en choisissant le bouton Enable (Activer).

Modification du rôle lié au service **AWSServiceRoleForAmazonSSM_OpsInsights** pour Systems Manager

Systems Manager ne vous permet pas de modifier le rôle lié à un service `AWSServiceRoleForAmazonSSM_OpsInsights`. Une fois que vous avez créé un rôle lié à un service, vous ne pouvez pas changer le nom du rôle, car plusieurs entités peuvent faire référence à ce rôle. Néanmoins, vous pouvez modifier la description du rôle à l'aide d'IAM. Pour en savoir plus, consultez Modification d'un rôle lié à un service dans le guide de l'utilisateur IAM.

Suppression du rôle lié au service **AWSServiceRoleForAmazonSSM_OpsInsights** 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 du rôle lié au service **AWSServiceRoleForAmazonSSM_OpsInsights**

Avant de pouvoir utiliser IAM pour supprimer un rôle lié au service `AWSServiceRoleForAmazonSSM_OpsInsights`, vous devez d'abord désactiver les informations opérationnelles dans Systems Manager OpsCenter. Pour de plus amples informations, veuillez consulter Analyse des informations opérationnelles pour réduire OpsItems.

Suppression manuelle du rôle lié au service **AWSServiceRoleForAmazonSSM_OpsInsights**

Utilisez la console IAM, l'AWS CLI ou l'API AWS pour supprimer le rôle lié à un service `AWSServiceRoleForAmazonSSM_OpsInsights`. Pour plus d'informations, consultez Suppression d'un rôle lié à un service dans le Guide de l'utilisateur IAM.

Régions prises en charge pour le rôle lié à un service **`AWSServiceRoleForAmazonSSM_OpsInsights`** de Systems Manager

Systems Manager ne prend pas en charge l'utilisation de rôles liés à un service dans toutes les régions où le service est disponible. Vous pouvez utiliser le rôle AWSServiceRoleForAmazonSSM_OpsInsights dans les Régions suivantes.

Nom de la région	Identité de la région	Prise en charge dans Systems Manager
US East (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 (Tokyo)	ap-northeast-1	Oui
Asie-Pacifique (Séoul)	ap-northeast-2	Oui
Asie-Pacifique (Singapore)	ap-southeast-1	Oui
Asie-Pacifique (Sydney)	ap-southeast-2	Oui
Asie-Pacifique (Hong Kong)	ap-east-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
Europe (Stockholm)	eu-north-1	Oui

Nom de la région	Identité de la région	Prise en charge dans Systems Manager
Europe (Milan)	eu-south-1	Oui
Amérique du Sud (São Paulo)	sa-east-1	Oui
Moyen-Orient (Bahreïn)	me-south-1	Oui
Afrique (Le Cap)	af-south-1	Oui
AWS GovCloud (US)	us-gov-west-1	Oui
AWS GovCloud (US)	us-gov-east-1	Oui

Utiliser des rôles pour exporter Explorer OpsData

AWS Systems Manager Explorerutilise le rôle de service de ExplorerExportrôle AmazonSSM pour exporter les données d'opérations (OpsData) à l'aide du runbook `AWS-ExportOpsDataToS3` d'automatisation.

Autorisations des rôles liés à un service pour Explorer

Le rôle lié à un service `AmazonSSMExplorerExportRole` fait confiance uniquement à `ssm.amazonaws.com` pour assumer le rôle.

Vous pouvez utiliser le rôle `AmazonSSMExplorerExportRole` lié à un service pour exporter les données d'opérations (OpsData) à l'aide du runbook `AWS-ExportOpsDataToS3` d'automatisation. Vous pouvez exporter 5 000 OpsData articles depuis Explorer un fichier de valeurs séparées par des virgules (.csv) vers un bucket Amazon Simple Storage Service (Amazon S3).

La politique d'autorisations liée au rôle permet à Systems Manager de réaliser les actions suivantes sur les ressources spécifiées :

- `s3:PutObject`
- `s3:GetBucketAcl`
- `s3:GetBucketLocation`
- `sns:Publish`

- `logs:DescribeLogGroups`
- `logs:DescribeLogStreams`
- `logs:CreateLogGroup`
- `logs:PutLogEvents`
- `logs:CreateLogStream`
- `ssm:GetOpsSummary`

Vous devez configurer les autorisations de manière à permettre à une entité IAM (comme un utilisateur, un groupe ou un rôle) de créer, modifier ou supprimer un rôle lié à un service. Pour plus d'informations, consultez Autorisations de rôles liés à un service dans le Guide de l'utilisateur IAM.

Création du rôle lié au service **`AmazonSSMExplorerExportRole`** pour Systems Manager

Systems Manager crée le rôle `AmazonSSMExplorerExportRole` lié au service lorsque vous exportez à OpsData l'aide de Explorer la console Systems Manager. Pour plus d'informations, consultez Exportation OpsData depuis Systems Manager Explorer.

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 du rôle lié au service **`AmazonSSMExplorerExportRole`** pour Systems Manager

Systems Manager ne vous permet pas de modifier le rôle lié à un service `AmazonSSMExplorerExportRole`. Une fois que vous avez créé un rôle lié à un service, vous ne pouvez pas changer le nom du rôle, car plusieurs entités peuvent faire référence au rôle. Néanmoins, vous pouvez modifier la description du rôle à l'aide d'IAM. Pour en savoir plus, consultez Modification d'un rôle lié à un service dans le guide de l'utilisateur IAM.

Suppression du rôle lié au service **`AmazonSSMExplorerExportRole`** 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, aucune entité inutilisée n'est surveillée ou gérée activement. Vous pouvez utiliser la console IAM AWS CLI, 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.

> **Note**
>
> Si le service Systems Manager utilise le rôle lorsque vous essayez de supprimer les balises ou les groupes de ressources, la suppression peut échouer. Si cela se produit, patientez quelques minutes et réessayez.

Pour supprimer les ressources Systems Manager utilisées par le service **`AmazonSSMExplorerExportRole`**

1. Pour supprimer des balises, consultez Ajout et suppression de balises sur une ressource individuelle.
2. Pour supprimer des groupes de ressources, voir Supprimer des groupes de AWS Resource Groups.

Pour supprimer manuellement le rôle lié au service **`AmazonSSMExplorerExportRole`** à l'aide d'IAM

Utilisez la console IAM AWS CLI, ou l'API IAM pour supprimer le rôle lié au `AmazonSSMExplorerExportRole` service. Pour plus d'informations, consultez Suppression d'un rôle lié à un service dans le Guide de l'utilisateur IAM.

Régions prises en charge pour le rôle lié à un service **`AmazonSSMExplorerExportRole`** de Systems Manager

Systems Managerprend en charge l'utilisation du rôle `AmazonSSMExplorerExportRole` lié au service partout Régions AWS où le service est disponible. Pour plus d'informations, consultez Points de terminaison et quotas AWS Systems Manager.

Journalisation et surveillance dans AWS Systems Manager

La surveillance joue un rôle important dans le maintien de la fiabilité, de la disponibilité AWS Systems Manager et des performances de vos AWS solutions. Vous devez collecter des données

de surveillance provenant de toutes les parties de votre AWS solution afin de pouvoir mieux corriger une défaillance multipoint, le cas échéant. AWS fournit plusieurs outils pour surveiller vos ressources Systems Manager et les autres ressources et répondre aux incidents potentiels.

AWS CloudTrail journaux

CloudTrail fournit un enregistrement des actions entreprises par un utilisateur, un rôle ou un Service AWS utilisateurSystems Manager. À l'aide des informations collectées par CloudTrail, vous pouvez déterminer la demande qui a été faiteSystems Manager, l'adresse IP à partir de laquelle la demande a été faite, qui a fait la demande, quand elle a été faite et des détails supplémentaires. Pour plus d'informations, consultez Journalisation des appels d' AWS Systems Manager API avec AWS CloudTrail.

CloudWatch Alarmes Amazon

À l'aide des CloudWatch alarmes Amazon, vous observez une seule métrique sur une période que vous spécifiez pour vos instances Amazon Elastic Compute Cloud (Amazon EC2) et pour d'autres ressources. Si la métrique dépasse un seuil donné, une notification est envoyée à un sujet ou à une politique Amazon Simple Notification Service (Amazon SNS). AWS Auto Scaling CloudWatch les alarmes n'appellent pas d'actions parce qu'elles sont dans un état particulier. L'état doit avoir changé et avoir été conservé pendant un nombre de périodes spécifié. Pour plus d'informations, consultez la section Utilisation des CloudWatch alarmes Amazon dans le guide de CloudWatch l'utilisateur Amazon.

Tableaux de CloudWatch bord Amazon

CloudWatch les tableaux de bord sont des pages d'accueil personnalisables dans la CloudWatch console que vous pouvez utiliser pour surveiller vos ressources dans une seule vue, même celles qui sont réparties entre différentes Régions AWS entités. Vous pouvez utiliser CloudWatch des tableaux de bord pour créer des vues personnalisées des mesures et des alarmes relatives à vos AWS ressources. Pour plus d'informations, consultez CloudWatchTableaux de bord Amazon hébergés par Systems Manager.

Amazon EventBridge

À l'aide d'Amazon EventBridge, vous pouvez configurer des règles pour vous avertir des modifications apportées aux Systems Manager ressources et EventBridge pour vous demander de prendre des mesures en fonction du contenu de ces événements. EventBridge prend en charge un certain nombre d'événements émis par diverses Systems Manager fonctionnalités. Pour plus d'informations, consultez Surveillance d'événements Systems Manager avec Amazon EventBridge.

Amazon CloudWatch Logs et SSM Agent journaux

L'SSM Agent écrit des informations relatives aux exécutions, actions planifiées, erreurs et états d'intégrité dans les fichiers journaux de chaque nœud. Vous pouvez afficher les fichiers journaux en vous connectant manuellement à un nœud. Nous recommandons d'envoyer automatiquement les données du journal de l'agent à un groupe de CloudWatch journaux dans Logs à des fins d'analyse. Pour plus d'informations, consultez Envoi des journaux des nœuds vers CloudWatch des journaux unifiés (CloudWatch agent) et Affichage des journaux SSM Agent.

Conformité d'AWS Systems Manager

Vous pouvez utiliser Compliance, une fonctionnalité de AWS Systems Manager, pour analyser votre parc de nœuds gérés afin de détecter la conformité des correctifs et les incohérences de configuration. Vous pouvez collecter et agréger des données provenant de plusieurs Comptes AWS sources Régions AWS, puis explorer des ressources spécifiques non conformes. Par défaut, Compliance affiche les données de conformité actuelles concernant l'application de correctifs Patch Manager AWS Systems Manager, une fonctionnalité et les associations dans State Manager une fonctionnalité de AWS Systems Manager. Pour plus d'informations, consultez Conformité d'AWS Systems Manager.

AWS Systems Manager Explorer

Explorer, une fonctionnalité de AWS Systems Manager, est un tableau de bord des opérations personnalisable qui fournit des informations sur vos AWS ressources. Exploreraffiche une vue agrégée des données d'exploitation (OpsData) pour vous Comptes AWS et pour l'ensemble de celles-ci Régions AWS. DansExplorer, OpsData inclut les métadonnées relatives à vos instances EC2, les détails de conformité des correctifs et les éléments de travail opérationnels (OpsItems). Explorerfournit un contexte sur la manière dont elles OpsItems sont réparties entre vos unités commerciales ou vos applications, sur leur évolution dans le temps et sur 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. Pour plus d'informations, consultez AWS Systems Manager Explorer.

AWS Systems Manager OpsCenter

OpsCenter, une fonctionnalité de AWS Systems Manager, fournit un emplacement central où les ingénieurs des opérations et les professionnels de l'informatique peuvent consulter, étudier et résoudre les éléments de travail opérationnels (OpsItems) liés aux AWS ressources. OpsCenteragrège et normalise OpsItems l'ensemble des services tout en fournissant des données d'investigation contextuelles sur chacun des servicesOpsItem, ainsi que sur les

ressources connexes OpsItems et connexes. OpsCenterfournit également des runbooks dans Automation, une fonctionnalité de AWS Systems Manager, que vous pouvez utiliser pour résoudre rapidement les problèmes. OpsCenterest intégré à Amazon EventBridge. Cela signifie que vous pouvez créer des EventBridge règles qui se créent automatiquement OpsItems pour tous ceux Service AWS qui publient des événements sur EventBridge. Pour plus d'informations, consultez AWS Systems Manager OpsCenter.

Amazon Simple Notification Service

Vous pouvez configurer Amazon Simple Notification Service (Amazon SNS) de sorte à envoyer des notifications sur le statut des commandes que vous envoyez à l'aide de Run Command ou Maintenance Windows, qui sont des fonctionnalités de AWS Systems Manager. Amazon SNS coordonne et gère la réception ou l'envoi de notifications aux points de terminaison ou aux clients abonnés aux rubriques Amazon SNS. Vous pouvez recevoir une notification chaque fois qu'une commande change de statut ou passe à un statut spécifique, par exemple, `Failed` ou `Timed Out`. Lorsque vous envoyez une commande à plusieurs nœuds, vous pouvez recevoir une notification pour chaque copie de la commande envoyée à un nœud spécifique. Pour plus d'informations, consultez Surveillance des changements d'état du Systems Manager à l'aide des notifications Amazon SNS.

AWS Trusted Advisor et AWS Health Dashboard

Trusted Advisor s'appuie sur les meilleures pratiques apprises en servant des centaines de milliers de AWS clients. Trusted Advisor inspecte votre AWS environnement, puis émet des recommandations lorsque des opportunités se présentent pour économiser de l'argent, améliorer la disponibilité et les performances du système ou contribuer à combler les failles de sécurité. Tous les AWS clients ont accès à cinq Trusted Advisor chèques. Les clients disposant d'un forfait AWS Support Business ou Enterprise peuvent consulter tous les Trusted Advisor chèques. Pour de plus amples informations, veuillez consulter AWS Trusted Advisor dans le Guide de l'utilisateur de AWS Support et le Guide de l'utilisateur de AWS Health.

Plus d'informations

- Surveillance AWS Systems Manager

Validation de la conformité pour AWS Systems Manager

Cette rubrique traite de la conformité d'AWS Systems Manager avec les programmes d'assurance tiers. Pour plus d'informations sur l'affichage des données de conformité pour vos nœuds gérés, consultez Conformité d'AWS Systems Manager.

Des auditeurs tiers évaluent la sécurité et la conformité de Systems Manager dans le cadre de plusieurs programmes de conformité AWS. Il s'agit notamment des certifications SOC, PCI, FedRAMP, HIPAA et d'autres.

Pour obtenir une liste des services Services AWS relevant de programmes de conformité spécifiques, reportez-vous à Services AWS relevant de programme de conformité. Pour obtenir des renseignements généraux, consultez Programmes de conformité AWS.

Vous pouvez télécharger les rapports de l'audit externe avec AWS Artifact. Pour plus d'informations, voir Téléchargement de rapports sur AWS Artifact.

Votre responsabilité de conformité lors de l'utilisation de Systems Manager est déterminée par la sensibilité de vos données, les objectifs de conformité de votre entreprise, ainsi que par la législation et la réglementation applicables. AWS fournit les ressources suivantes pour faciliter le respect de la conformité :

- Guides de Quick Start (démarrage rapide) de la sécurité et de la conformité– Ces guides de déploiement traitent des considérations architecturales et fournissent des étapes pour déployer des environnements de base axés sur la sécurité et la conformité sur AWS.
- Livre blanc sur l'architecture pour la sécurité et la conformité HIPAA : ce livre blanc décrit comment les entreprises peuvent utiliser AWS pour créer des applications conformes à HIPAA.
- Ressources de conformité AWS – Cet ensemble de manuels et de guides peut s'appliquer à votre secteur d'activité et à votre emplacement.
- Évaluation des ressources à l'aide de règles dans le Guide 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, ce qui vous permet de vérifier votre conformité aux normes du secteur et aux bonnes pratiques de sécurité.

Résilience dans AWS Systems Manager

L'infrastructure mondiale d'AWS est construite autour de zones de disponibilité et de Régions AWS. Les Régions AWSfournissent 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

davantage disponibles, tolérantes aux pannes et ont une plus grande capacité de mise à l'échelle que les infrastructures traditionnelles à un ou plusieurs centres de données.

Pour plus d'informations sur les Régions AWS et les zones de disponibilité, consultez 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 mondial AWS. Pour plus d'informations sur les services de sécurité AWS et la manière dont AWS protège l'infrastructure, consultez la section Sécurité du cloud AWS. Pour concevoir votre environnement AWS en utilisant les meilleures pratiques en matière de sécurité de l'infrastructure, consultez la section Protection de l'infrastructure dans le Security Pillar AWS Well-Architected Framework (Pilier de sécurité de l'infrastructure Well-Architected Framework).

Vous utilisez les appels d'API publiés AWS pour accéder à Systems Manager via le réseau. Les clients doivent prendre en charge les éléments suivants :

- Protocole TLS (Transport Layer Security). Nous exigeons TLS 1.2 et nous recommandons TLS 1.3.
- Ses suites de chiffrement PFS (Perfect Forward Secrecy) comme DHE (Ephemeral Diffie-Hellman) ou ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). La plupart des systèmes modernes tels que Java 7 et les versions ultérieures prennent en charge ces modes.

En outre, les demandes doivent être signées à l'aide d'un ID de clé d'accès et d'une clé d'accès secrète associée à un principal IAM. Vous pouvez également utiliser AWS Security Token Service (AWS STS) pour générer des informations 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 telles que la configuration du pare-feu et la reprise après sinistre. Ces procédures ont été vérifiées et certifiées par les tiers appropriés. Pour de plus amples informations, consultez les ressources suivantes.

- Validation de la conformité pour AWS Systems Manager

- Modèle de responsabilité partagée
- Bonnes pratiques en matière de sécurité, d'identité et de conformité

Bonnes pratiques de sécurité pour Systems Manager

AWS Systems Manager fournit un certain nombre de fonctionnalités de sécurité à prendre en compte lors de l'élaboration et de la mise en œuvre de vos propres politiques 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
- Bonnes pratiques de surveillance et d'audit pour Systems Manager

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 sélectionnez qui obtient les autorisations pour telles ou telles ressources Systems Manager. Vous autorisez 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 :

- Politiques IAM et Limites d'autorisations pour des entités IAM
- Politiques de contrôle des services

Utiliser les paramètres recommandés SSM Agent lorsque vous êtes configuré pour utiliser un proxy

Si vous configurez SSM Agent pour utiliser un proxy, utilisez la `no_proxy` variable avec l'adresse IP du service de métadonnées de l'instance de Systems Manager pour vous assurer que les appels à Systems Manager ne prennent pas l'identité du service proxy.

Pour plus d'informations, consultez Configuration SSM Agent pour utiliser un proxy sur les nœuds Linux et Configurer l'SSM Agent pour utiliser un proxy pour les instances Windows Server.

Utiliser SecureString des paramètres pour chiffrer et protéger les données secrètes

DansParameter Store, une capacité de AWS Systems Manager, un `SecureString` paramètre désigne toute donnée sensible qui doit être stockée et référencée de manière sécurisée. Si vous ne souhaitez pas que les utilisateurs modifient ou référencent en texte brut, telles que des mots de passe ou des clés de licence, créez ces paramètres à l'aide du type de `SecureString` données. Parameter Storeutilise un AWS KMS key in AWS Key Management Service (AWS KMS) pour chiffrer la valeur du paramètre. AWS KMS utilise soit une clé gérée par le client, soit une clé Clé gérée par AWS lors du chiffrement de la valeur du paramètre. Pour une sécurité maximale, nous vous recommandons d'utiliser votre propre clé KMS. Si vous utilisez le Clé gérée par AWS, tout utilisateur autorisé à exécuter les GetParametersactions GetParameteret dans votre compte peut consulter ou récupérer le contenu de tous les `SecureString` paramètres. Si vous utilisez des clés gérées par le client pour chiffrer vos valeurs `SecureString` sécurisées, vous pouvez utiliser des politiques IAM et des politiques de clé pour gérer les autorisations de chiffrement et de déchiffrement des paramètres. Il est plus difficile d'établir des politiques de contrôle d'accès pour ces opérations lorsque vous utilisez les clés gérées par le client. Par exemple, si vous utilisez le Clé gérée par AWS pour chiffrer des `SecureString` paramètres et que vous ne souhaitez pas que les utilisateurs utilisent des `SecureString` paramètres, leurs politiques IAM doivent explicitement refuser l'accès à la clé par défaut.

Pour de plus amples informations, consultez Restriction de l'accès aux paramètres Systems Manager à l'aide des politiques IAM et Comment AWS Systems ManagerParameter Store utilise la AWS KMS dans le Manuel du développeur AWS Key Management Service .

Définir des allowedValues et un allowedPattern pour les paramètres de document

Vous pouvez valider les entrées utilisateur pour les paramètres de documents Systems Manager (SSM) 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 Éléments de données et paramètres.

Bloquer le partage public de documents

À moins que votre cas d'utilisation exige que le partage public soit autorisé, nous vous recommandons d'activer le paramètre de partage public de bloc pour vos documents SSM dans la section Preferences (Préférences) de la console Systems Manager Documents.

Utilisation d'un Amazon Virtual Private Cloud (Amazon VPC) et de points de terminaison de VPC

Vous pouvez utiliser Amazon VPC pour lancer AWS des ressources 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 VPC, vous pouvez connecter de manière privée votre VPC aux services de point de terminaison VPC pris en charge et Services AWS alimentés par celui-ci AWS PrivateLink sans avoir besoin d'une passerelle Internet, d'un périphérique NAT, d'une connexion VPN ou d'une connexion. AWS Direct Connect 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 d'Amazon VPC, consultez Améliorer la sécurité des instances EC2 en utilisant les points de terminaison VPC pour Systems Manager et la confidentialité du trafic interréseau dans Amazon VPC dans le guide de l'utilisateur Amazon VPC.

Limiter les utilisateurs Session Manager aux sessions utilisant des commandes interactives et des documents de session SSM spécifiques

Session Manager, une des fonctionnalités de AWS Systems Manager, fournit plusieurs méthodes pour démarrer des sessions sur vos nœuds gérés. 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 et non interactives).

Pour plus de sécurité, vous pouvez limiter l'accès à Session Manager à des instances Amazon EC2 spécifiques et à des documents de session Session Manager spécifiques. Vous accordez ou révoquez Session Manager l'accès de cette manière en utilisant des politiques AWS Identity and Access Management (IAM). Pour plus d'informations, consultez Étape 3 : Contrôler les accès de session aux nœuds gérés.

Mise à disposition d'autorisations de nœud temporaires aux flux de travail Automation

Lors d'un flux de travail dans Automation, une des fonctionnalités de AWS Systems Manager, vos nœuds peuvent avoir besoin d'autorisations nécessaires pour cette exécution uniquement, mais pas pour d'autres opérations Systems Manager. Par exemple, un flux de travail d'automatisation peut nécessiter qu'un nœud appelle une opération d'API particulière ou accède à une AWS ressource 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 nœuds dans le runbook Automation lui-même au lieu d'ajouter les autorisations à votre profil d'instance IAM. À la fin du flux de travail 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 Automations sur l'AWS Management and Governance Blog.

Maintien à jour AWS et mise à jour des Systems Manager outils

AWS publie régulièrement des versions mises à jour d'outils et de plugins que vous pouvez utiliser dans le cadre de vos Systems Manager opérations AWS et de vos activités. La mise à jour de ces ressources garantit que les utilisateurs et les nœuds de votre compte ont accès aux fonctionnalités et aux fonctions de sécurité les plus récentes de ces outils.

- SSM Agent – AWS Systems Manager Agent (SSM Agent) est un logiciel Amazon qui peut être installé et configuré sur une instance Amazon Elastic Compute Cloud (Amazon EC2), un serveur sur site ou une machine virtuelle (VM). SSM Agent 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 Automatisation des mises à jour de l'SSM Agent. Nous vous recommandons également de vérifier la signature de SSM Agent dans le cadre de votre processus de mise à jour. Pour plus d'informations, veuillez consulter Vérification de la signature de SSM Agent.
- AWS CLI — The AWS Command Line Interface (AWS CLI) est un outil open source qui vous permet d'interagir à Services AWS l'aide de commandes dans votre interface de ligne de commande. Pour mettre à jour le AWS CLI, vous devez exécuter la même commande que celle utilisée pour installer le 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 la section Installation de la AWS CLI version 2 dans le guide de AWS Command Line Interface l'utilisateur.

- AWS Tools for Windows PowerShell — Les outils pour Windows PowerShell sont un ensemble de PowerShell modules basés sur les fonctionnalités proposées par le AWS SDK pour .NET. Ils vous AWS Tools for Windows PowerShell permettent de scripter des opérations sur vos AWS ressources à partir de la ligne de PowerShell commande. Régulièrement, à mesure que des versions mises à jour des Outils pour Windows PowerShell sont publiées, vous devez mettre à jour la version que vous exécutez localement. Pour plus d'informations, consultez la section Mise à AWS Tools for Windows PowerShell jour du sous Windows ou Mise à AWS Tools for Windows PowerShell jour du sous Linux ou macOS dans le guide de l'utilisateur d'IAM Policy Simulator.
- Plugin Session Manager – si les utilisateurs de votre organisation disposant des autorisations nécessaires pour utiliser Session Manager veulent se connecter à un nœud à l'aide de l' AWS CLI, ils doivent d'abord installer le plugin Session Manager 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 Installez le Session Manager plugin pour AWS CLI.
- CloudWatch agent : vous pouvez configurer et utiliser l' CloudWatch agent pour collecter des métriques et des journaux à partir de vos instances EC2, de vos instances sur site et de vos 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 AWS Systems Manager Quick Setup.

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 l'intégralité de vos resources Systems Manager

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 Recherche de ressources à baliser dans le Guide de l'utilisateur AWS Resource Groups .

Créez des groupes de ressources pour vos ressourcesSystems Manager. Pour plus d'informations, consultez Présentation des groupes de ressources.

Mettre en œuvre la surveillance à l'aide CloudWatch des outils de surveillance Amazon

La surveillance est un enjeu important pour assurer la fiabilité, la sécurité, la disponibilité et les performances d'Systems Manager et de vos solutions AWS . Amazon CloudWatch fournit plusieurs outils et services pour vous aider à surveiller Systems Manager et à surveiller vos autres Services AWS. Pour plus d'informations, consultez Envoi des journaux des nœuds vers CloudWatch des journaux unifiés (CloudWatch agent) et Surveillance d'événements Systems Manager avec Amazon EventBridge.

Utiliser CloudTrail

AWS CloudTrail fournit un enregistrement des actions entreprises par un utilisateur, un rôle ou un Service AWS utilisateurSystems Manager. À l'aide des informations collectées par CloudTrail, vous pouvez déterminer la demande qui a été faiteSystems Manager, l'adresse IP à partir de laquelle la demande a été faite, qui a fait la demande, quand elle a été faite et des détails supplémentaires. Pour plus d'informations, consultez Journalisation des appels d' AWS Systems Manager API avec AWS CloudTrail.

Allumez AWS Config

AWS Config vous permet d'évaluer, d'auditer et d'évaluer les configurations de vos AWS ressources. AWS Config surveille les configurations des ressources, ce qui vous permet d'évaluer les configurations enregistrées par rapport aux configurations sécurisées requises. Vous pouvez ainsi examiner les modifications apportées aux configurations et les relations entre les AWS ressources, étudier les historiques détaillés de configuration des ressources et déterminer votre conformité globale par rapport aux configurations spécifiées dans vos directives internes. AWS Config 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.

Surveillez les avis AWS de sécurité

Vous devriez consulter régulièrement les avis de sécurité publiés Trusted Advisor pour votre Compte AWS. Vous pouvez le faire par programmation en utilisant describe-trusted-advisor-checks.

De plus, surveillez activement l'adresse e-mail principale enregistrée pour chacun de vos Comptes AWS. AWS vous contactera, à l'aide de cette adresse e-mail, au sujet des problèmes de sécurité émergents susceptibles de vous affecter.

AWS les problèmes opérationnels ayant un impact important sont publiés sur le AWS Service Health Dashboard. Les problèmes opérationnels sont également publiés dans les différents comptes via le tableau de bord d'état personnel. Pour de plus amples d'informations, consultez la documentation AWS Health.

Plus d'informations

- Bonnes pratiques en matière de sécurité, d'identité et de conformité
- Mise en route : suivez les meilleures pratiques de sécurité lors de la configuration de vos AWS ressources (blog sur AWS la sécurité)
- Bonnes pratiques de sécurité dans IAM
- Bonnes pratiques en matière de sécurité dans AWS CloudTrail
- Bonnes pratiques de sécurité pour Simple Storage Service (Amazon S3)
- Bonnes pratiques en matière de sécurité pour AWS Key Management Service

Exemples de code pour Systems Manager utilisant des AWS SDK

Les exemples de code suivants montrent comment utiliser Systems Manager avec un kit de développement AWS logiciel (SDK).

Les actions sont des extraits de code de programmes plus larges et doivent être exécutées dans leur contexte. Alors que les actions vous indiquent comment appeler des fonctions de service individuelles, vous pouvez les voir en contexte dans leurs scénarios associés et dans des exemples interservices.

Les Scénarios sont des exemples de code qui vous montrent comment accomplir une tâche spécifique en appelant plusieurs fonctions au sein d'un même service.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Mise en route

Hello Systems Manager

L'exemple de code suivant montre comment commencer à utiliser Systems Manager.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.ssm.SsmClient;
import software.amazon.awssdk.services.ssm.model.DocumentFilter;
import software.amazon.awssdk.services.ssm.model.ListDocumentsRequest;
import software.amazon.awssdk.services.ssm.model.ListDocumentsResponse;
```

```
public class HelloSSM {

    public static void main(String[] args) {
        final String usage = """

                Usage:
                    <awsAccount>

                Where:
                    awsAccount - Your AWS Account number.
                """;

        if (args.length != 1) {
            System.out.println(usage);
            System.exit(1);
        }

        String awsAccount = args[0] ;
        Region region = Region.US_EAST_1;
        SsmClient ssmClient = SsmClient.builder()
            .region(region)
            .build();

        listDocuments(ssmClient, awsAccount);
    }

    /*
    This code automatically fetches the next set of results using the `nextToken`
 and
    stops once the desired maxResults (20 in this case) have been reached.
    */
    public static void listDocuments(SsmClient ssmClient, String awsAccount) {
        String nextToken = null;
        int totalDocumentsReturned = 0;
        int maxResults = 20;
        do {
            ListDocumentsRequest request = ListDocumentsRequest.builder()
                .documentFilterList(
                    DocumentFilter.builder()
                        .key("Owner")
                        .value(awsAccount)
                        .build()
                    )
                .maxResults(maxResults)
```

```
                .nextToken(nextToken)
                .build();

            ListDocumentsResponse response = ssmClient.listDocuments(request);
            response.documentIdentifiers().forEach(identifier ->
 System.out.println("Document Name: " + identifier.name()));
            nextToken = response.nextToken();
            totalDocumentsReturned += response.documentIdentifiers().size();
        } while (nextToken != null && totalDocumentsReturned < maxResults);
    }
}
```

- Pour plus de détails sur l'API, voir ListThings dans le manuel de référence des AWS SDK for Java 2.x API.

Exemples de code

- Actions pour Systems Manager utilisant des AWS SDK
  - Utilisation AddTagsToResource avec un AWS SDK ou une CLI
  - Utilisation CancelCommand avec un AWS SDK ou une CLI
  - Utilisation CreateActivation avec un AWS SDK ou une CLI
  - Utilisation CreateAssociation avec un AWS SDK ou une CLI
  - Utilisation CreateAssociationBatch avec un AWS SDK ou une CLI
  - Utilisation CreateDocument avec un AWS SDK ou une CLI
  - Utilisation CreateMaintenanceWindow avec un AWS SDK ou une CLI
  - Utilisation CreateOpsItem avec un AWS SDK ou une CLI
  - Utilisation CreatePatchBaseline avec un AWS SDK ou une CLI
  - Utilisation DeleteActivation avec un AWS SDK ou une CLI
  - Utilisation DeleteAssociation avec un AWS SDK ou une CLI
  - Utilisation DeleteDocument avec un AWS SDK ou une CLI
  - Utilisation DeleteMaintenanceWindow avec un AWS SDK ou une CLI
  - Utilisation DeleteParameter avec un AWS SDK ou une CLI
  - Utilisation DeletePatchBaseline avec un AWS SDK ou une CLI
  - Utilisation DeregisterManagedInstance avec un AWS SDK ou une CLI
  - Utilisation DeregisterPatchBaselineForPatchGroup avec un AWS SDK ou une CLI

- Utilisation DeregisterTargetFromMaintenanceWindow avec un AWS SDK ou une CLI
- Utilisation DeregisterTaskFromMaintenanceWindow avec un AWS SDK ou une CLI
- Utilisation DescribeActivations avec un AWS SDK ou une CLI
- Utilisation DescribeAssociation avec un AWS SDK ou une CLI
- Utilisation DescribeAssociationExecutionTargets avec un AWS SDK ou une CLI
- Utilisation DescribeAssociationExecutions avec un AWS SDK ou une CLI
- Utilisation DescribeAutomationExecutions avec un AWS SDK ou une CLI
- Utilisation DescribeAutomationStepExecutions avec un AWS SDK ou une CLI
- Utilisation DescribeAvailablePatches avec un AWS SDK ou une CLI
- Utilisation DescribeDocument avec un AWS SDK ou une CLI
- Utilisation DescribeDocumentPermission avec un AWS SDK ou une CLI
- Utilisation DescribeEffectiveInstanceAssociations avec un AWS SDK ou une CLI
- Utilisation DescribeEffectivePatchesForPatchBaseline avec un AWS SDK ou une CLI
- Utilisation DescribeInstanceAssociationsStatus avec un AWS SDK ou une CLI
- Utilisation DescribeInstanceInformation avec un AWS SDK ou une CLI
- Utilisation DescribeInstancePatchStates avec un AWS SDK ou une CLI
- Utilisation DescribeInstancePatchStatesForPatchGroup avec un AWS SDK ou une CLI
- Utilisation DescribeInstancePatches avec un AWS SDK ou une CLI
- Utilisation DescribeMaintenanceWindowExecutionTaskInvocations avec un AWS SDK ou une CLI
- Utilisation DescribeMaintenanceWindowExecutionTasks avec un AWS SDK ou une CLI
- Utilisation DescribeMaintenanceWindowExecutions avec un AWS SDK ou une CLI
- Utilisation DescribeMaintenanceWindowTargets avec un AWS SDK ou une CLI
- Utilisation DescribeMaintenanceWindowTasks avec un AWS SDK ou une CLI
- Utilisation DescribeMaintenanceWindows avec un AWS SDK ou une CLI
- Utilisation DescribeOpsItems avec un AWS SDK ou une CLI
- Utilisation DescribeParameters avec un AWS SDK ou une CLI
- Utilisation DescribePatchBaselines avec un AWS SDK ou une CLI
- Utilisation DescribePatchGroupState avec un AWS SDK ou une CLI
- Utilisation DescribePatchGroups avec un AWS SDK ou une CLI

- Utilisation GetAutomationExecution avec un AWS SDK ou une CLI
- Utilisation GetCommandInvocation avec un AWS SDK ou une CLI
- Utilisation GetConnectionStatus avec un AWS SDK ou une CLI
- Utilisation GetDefaultPatchBaseline avec un AWS SDK ou une CLI
- Utilisation GetDeployablePatchSnapshotForInstance avec un AWS SDK ou une CLI
- Utilisation GetDocument avec un AWS SDK ou une CLI
- Utilisation GetInventory avec un AWS SDK ou une CLI
- Utilisation GetInventorySchema avec un AWS SDK ou une CLI
- Utilisation GetMaintenanceWindow avec un AWS SDK ou une CLI
- Utilisation GetMaintenanceWindowExecution avec un AWS SDK ou une CLI
- Utilisation GetMaintenanceWindowExecutionTask avec un AWS SDK ou une CLI
- Utilisation GetParameterHistory avec un AWS SDK ou une CLI
- Utilisation GetParameters avec un AWS SDK ou une CLI
- Utilisation GetPatchBaseline avec un AWS SDK ou une CLI
- Utilisation GetPatchBaselineForPatchGroup avec un AWS SDK ou une CLI
- Utilisation ListAssociationVersions avec un AWS SDK ou une CLI
- Utilisation ListAssociations avec un AWS SDK ou une CLI
- Utilisation ListCommandInvocations avec un AWS SDK ou une CLI
- Utilisation ListCommands avec un AWS SDK ou une CLI
- Utilisation ListComplianceItems avec un AWS SDK ou une CLI
- Utilisation ListComplianceSummaries avec un AWS SDK ou une CLI
- Utilisation ListDocumentVersions avec un AWS SDK ou une CLI
- Utilisation ListDocuments avec un AWS SDK ou une CLI
- Utilisation ListInventoryEntries avec un AWS SDK ou une CLI
- Utilisation ListResourceComplianceSummaries avec un AWS SDK ou une CLI
- Utilisation ListTagsForResource avec un AWS SDK ou une CLI
- Utilisation ModifyDocumentPermission avec un AWS SDK ou une CLI
- Utilisation PutComplianceItems avec un AWS SDK ou une CLI
- Utilisation PutInventory avec un AWS SDK ou une CLI
- Utilisation PutParameter avec un AWS SDK ou une CLI

- Utilisation RegisterDefaultPatchBaseline avec un AWS SDK ou une CLI
- Utilisation RegisterPatchBaselineForPatchGroup avec un AWS SDK ou une CLI
- Utilisation RegisterTargetWithMaintenanceWindow avec un AWS SDK ou une CLI
- Utilisation RegisterTaskWithMaintenanceWindow avec un AWS SDK ou une CLI
- Utilisation RemoveTagsFromResource avec un AWS SDK ou une CLI
- Utilisation SendCommand avec un AWS SDK ou une CLI
- Utilisation StartAutomationExecution avec un AWS SDK ou une CLI
- Utilisation StopAutomationExecution avec un AWS SDK ou une CLI
- Utilisation UpdateAssociation avec un AWS SDK ou une CLI
- Utilisation UpdateAssociationStatus avec un AWS SDK ou une CLI
- Utilisation UpdateDocument avec un AWS SDK ou une CLI
- Utilisation UpdateDocumentDefaultVersion avec un AWS SDK ou une CLI
- Utilisation UpdateMaintenanceWindow avec un AWS SDK ou une CLI
- Utilisation UpdateManagedInstanceRole avec un AWS SDK ou une CLI
- Utilisation UpdateOpsItem avec un AWS SDK ou une CLI
- Utilisation UpdatePatchBaseline avec un AWS SDK ou une CLI

- Scénarios pour Systems Manager utilisant des AWS SDK
  - Commencez à utiliser Systems Manager à l'aide d'un AWS SDK

Actions pour Systems Manager utilisant des AWS SDK

Les exemples de code suivants montrent comment effectuer des actions individuelles de Systems Manager avec des AWS SDK. Ces extraits appellent l'API Systems Manager et sont des extraits de code de programmes plus volumineux qui doivent être exécutés en contexte. Chaque exemple inclut un lien vers GitHub, où vous pouvez trouver des instructions pour configurer et exécuter le code.

Les exemples suivants incluent uniquement les actions les plus couramment utilisées. Pour obtenir la liste complète, veuillez consulter la AWS Systems Manager Référence d'API.

Exemples

- Utilisation AddTagsToResource avec un AWS SDK ou une CLI
- Utilisation CancelCommand avec un AWS SDK ou une CLI
- Utilisation CreateActivation avec un AWS SDK ou une CLI

- Utilisation CreateAssociation avec un AWS SDK ou une CLI
- Utilisation CreateAssociationBatch avec un AWS SDK ou une CLI
- Utilisation CreateDocument avec un AWS SDK ou une CLI
- Utilisation CreateMaintenanceWindow avec un AWS SDK ou une CLI
- Utilisation CreateOpsItem avec un AWS SDK ou une CLI
- Utilisation CreatePatchBaseline avec un AWS SDK ou une CLI
- Utilisation DeleteActivation avec un AWS SDK ou une CLI
- Utilisation DeleteAssociation avec un AWS SDK ou une CLI
- Utilisation DeleteDocument avec un AWS SDK ou une CLI
- Utilisation DeleteMaintenanceWindow avec un AWS SDK ou une CLI
- Utilisation DeleteParameter avec un AWS SDK ou une CLI
- Utilisation DeletePatchBaseline avec un AWS SDK ou une CLI
- Utilisation DeregisterManagedInstance avec un AWS SDK ou une CLI
- Utilisation DeregisterPatchBaselineForPatchGroup avec un AWS SDK ou une CLI
- Utilisation DeregisterTargetFromMaintenanceWindow avec un AWS SDK ou une CLI
- Utilisation DeregisterTaskFromMaintenanceWindow avec un AWS SDK ou une CLI
- Utilisation DescribeActivations avec un AWS SDK ou une CLI
- Utilisation DescribeAssociation avec un AWS SDK ou une CLI
- Utilisation DescribeAssociationExecutionTargets avec un AWS SDK ou une CLI
- Utilisation DescribeAssociationExecutions avec un AWS SDK ou une CLI
- Utilisation DescribeAutomationExecutions avec un AWS SDK ou une CLI
- Utilisation DescribeAutomationStepExecutions avec un AWS SDK ou une CLI
- Utilisation DescribeAvailablePatches avec un AWS SDK ou une CLI
- Utilisation DescribeDocument avec un AWS SDK ou une CLI
- Utilisation DescribeDocumentPermission avec un AWS SDK ou une CLI
- Utilisation DescribeEffectiveInstanceAssociations avec un AWS SDK ou une CLI
- Utilisation DescribeEffectivePatchesForPatchBaseline avec un AWS SDK ou une CLI
- Utilisation DescribeInstanceAssociationsStatus avec un AWS SDK ou une CLI
- Utilisation DescribeInstanceInformation avec un AWS SDK ou une CLI

- Utilisation DescribeInstancePatchStates avec un AWS SDK ou une CLI
- Utilisation DescribeInstancePatchStatesForPatchGroup avec un AWS SDK ou une CLI
- Utilisation DescribeInstancePatches avec un AWS SDK ou une CLI
- Utilisation DescribeMaintenanceWindowExecutionTaskInvocations avec un AWS SDK ou une CLI
- Utilisation DescribeMaintenanceWindowExecutionTasks avec un AWS SDK ou une CLI
- Utilisation DescribeMaintenanceWindowExecutions avec un AWS SDK ou une CLI
- Utilisation DescribeMaintenanceWindowTargets avec un AWS SDK ou une CLI
- Utilisation DescribeMaintenanceWindowTasks avec un AWS SDK ou une CLI
- Utilisation DescribeMaintenanceWindows avec un AWS SDK ou une CLI
- Utilisation DescribeOpsItems avec un AWS SDK ou une CLI
- Utilisation DescribeParameters avec un AWS SDK ou une CLI
- Utilisation DescribePatchBaselines avec un AWS SDK ou une CLI
- Utilisation DescribePatchGroupState avec un AWS SDK ou une CLI
- Utilisation DescribePatchGroups avec un AWS SDK ou une CLI
- Utilisation GetAutomationExecution avec un AWS SDK ou une CLI
- Utilisation GetCommandInvocation avec un AWS SDK ou une CLI
- Utilisation GetConnectionStatus avec un AWS SDK ou une CLI
- Utilisation GetDefaultPatchBaseline avec un AWS SDK ou une CLI
- Utilisation GetDeployablePatchSnapshotForInstance avec un AWS SDK ou une CLI
- Utilisation GetDocument avec un AWS SDK ou une CLI
- Utilisation GetInventory avec un AWS SDK ou une CLI
- Utilisation GetInventorySchema avec un AWS SDK ou une CLI
- Utilisation GetMaintenanceWindow avec un AWS SDK ou une CLI
- Utilisation GetMaintenanceWindowExecution avec un AWS SDK ou une CLI
- Utilisation GetMaintenanceWindowExecutionTask avec un AWS SDK ou une CLI
- Utilisation GetParameterHistory avec un AWS SDK ou une CLI
- Utilisation GetParameters avec un AWS SDK ou une CLI
- Utilisation GetPatchBaseline avec un AWS SDK ou une CLI
- Utilisation GetPatchBaselineForPatchGroup avec un AWS SDK ou une CLI
- Utilisation ListAssociationVersions avec un AWS SDK ou une CLI

- Utilisation ListAssociations avec un AWS SDK ou une CLI
- Utilisation ListCommandInvocations avec un AWS SDK ou une CLI
- Utilisation ListCommands avec un AWS SDK ou une CLI
- Utilisation ListComplianceItems avec un AWS SDK ou une CLI
- Utilisation ListComplianceSummaries avec un AWS SDK ou une CLI
- Utilisation ListDocumentVersions avec un AWS SDK ou une CLI
- Utilisation ListDocuments avec un AWS SDK ou une CLI
- Utilisation ListInventoryEntries avec un AWS SDK ou une CLI
- Utilisation ListResourceComplianceSummaries avec un AWS SDK ou une CLI
- Utilisation ListTagsForResource avec un AWS SDK ou une CLI
- Utilisation ModifyDocumentPermission avec un AWS SDK ou une CLI
- Utilisation PutComplianceItems avec un AWS SDK ou une CLI
- Utilisation PutInventory avec un AWS SDK ou une CLI
- Utilisation PutParameter avec un AWS SDK ou une CLI
- Utilisation RegisterDefaultPatchBaseline avec un AWS SDK ou une CLI
- Utilisation RegisterPatchBaselineForPatchGroup avec un AWS SDK ou une CLI
- Utilisation RegisterTargetWithMaintenanceWindow avec un AWS SDK ou une CLI
- Utilisation RegisterTaskWithMaintenanceWindow avec un AWS SDK ou une CLI
- Utilisation RemoveTagsFromResource avec un AWS SDK ou une CLI
- Utilisation SendCommand avec un AWS SDK ou une CLI
- Utilisation StartAutomationExecution avec un AWS SDK ou une CLI
- Utilisation StopAutomationExecution avec un AWS SDK ou une CLI
- Utilisation UpdateAssociation avec un AWS SDK ou une CLI
- Utilisation UpdateAssociationStatus avec un AWS SDK ou une CLI
- Utilisation UpdateDocument avec un AWS SDK ou une CLI
- Utilisation UpdateDocumentDefaultVersion avec un AWS SDK ou une CLI
- Utilisation UpdateMaintenanceWindow avec un AWS SDK ou une CLI
- Utilisation UpdateManagedInstanceRole avec un AWS SDK ou une CLI
- Utilisation UpdateOpsItem avec un AWS SDK ou une CLI
- Utilisation UpdatePatchBaseline avec un AWS SDK ou une CLI

Utilisation **AddTagsToResource** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliserAddTagsToResource.

CLI

AWS CLI

Exemple 1 : pour ajouter des balises à une fenêtre de maintenance

L'add-tags-to-resourceexemple suivant ajoute une balise à la fenêtre de maintenance spécifiée.

```
aws ssm add-tags-to-resource \
    --resource-type "MaintenanceWindow" \
    --resource-id "mw-03eb9db428EXAMPLE" \
    --tags "Key=Stack,Value=Production"
```

Cette commande ne produit aucun résultat.

Exemple 2 : pour ajouter des balises à un paramètre

L'add-tags-to-resourceexemple suivant ajoute deux balises au paramètre spécifié.

```
aws ssm add-tags-to-resource \
    --resource-type "Parameter" \
    --resource-id "My-Parameter" \
    --tags '[{"Key":"Region","Value":"East"},{"Key":"Environment",
 "Value":"Production"}]'
```

Cette commande ne produit aucun résultat.

Exemple 3 : pour ajouter des balises à un document SSM

L'add-tags-to-resourceexemple suivant ajoute une balise au document spécifié.

```
aws ssm add-tags-to-resource \
    --resource-type "Document" \
    --resource-id "My-Document" \
    --tags "Key=Quarter,Value=Q322"
```

Cette commande ne produit aucun résultat.

Pour plus d'informations, consultez les ressources de Tagging Systems Manager dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous AddTagsToResourceà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple met à jour une fenêtre de maintenance avec de nouvelles balises. Il n'y a pas de sortie si la commande réussit. La syntaxe utilisée dans cet exemple nécessite PowerShell la version 3 ou ultérieure.

```
$option1 = @{Key="Stack";Value=@("Production")}
Add-SSMResourceTag -ResourceId "mw-03eb9db42890fb82d" -ResourceType
 "MaintenanceWindow" -Tag $option1
```

Exemple 2 : Avec PowerShell la version 2, vous devez utiliser New-Object pour créer chaque balise. Il n'y a aucune sortie si la commande aboutit.

```
$tag1 = New-Object Amazon.SimpleSystemsManagement.Model.Tag
$tag1.Key = "Stack"
$tag1.Value = "Production"

Add-SSMResourceTag -ResourceId "mw-03eb9db42890fb82d" -ResourceType
 "MaintenanceWindow" -Tag $tag1
```

- Pour plus de détails sur l'API, reportez-vous AddTagsToResourceà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`CancelCommand`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`CancelCommand`.

CLI

AWS CLI

Exemple 1 : annuler une commande pour toutes les instances

L'`cancel-command`exemple suivant tente d'annuler la commande spécifiée qui est déjà en cours d'exécution pour toutes les instances.

```
aws ssm cancel-command \
    --command-id "662add3d-5831-4a10-b64a-f2ff3EXAMPLE"
```

Cette commande ne produit aucun résultat.

Exemple 2 : pour annuler une commande pour des instances spécifiques

L'`cancel-command`exemple suivant tente d'annuler une commande pour l'instance spécifiée uniquement.

```
aws ssm cancel-command \
    --command-id "662add3d-5831-4a10-b64a-f2ff3EXAMPLE"
    --instance-ids "i-02573cafcfEXAMPLE"
```

Cette commande ne produit aucun résultat.

Pour plus d'informations, voir Tagging Systems Manager Parameters dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous CancelCommandà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple tente d'annuler une commande. Il n'y a aucune sortie si l'opération réussit.

```
Stop-SSMCommand -CommandId "9ded293e-e792-4440-8e3e-7b8ec5feaa38"
```

- Pour plus de détails sur l'API, reportez-vous CancelCommandà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`CreateActivation`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`CreateActivation`.

CLI

AWS CLI

Pour créer une activation d'instance gérée

L'`create-activation`exemple suivant crée une activation d'instance gérée.

```
aws ssm create-activation \
    --default-instance-name "HybridWebServers" \
    --iam-role "HybridWebServersRole" \
    --registration-limit 5
```

Sortie :

```
{
    "ActivationId": "5743558d-563b-4457-8682-d16c3EXAMPLE",
    "ActivationCode": "dRmgnYaFv567vEXAMPLE"
}
```

Pour plus d'informations, consultez Étape 4 : Création d'une activation d'instance gérée pour un environnement hybride dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous CreateActivationà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple crée une instance gérée.

```
New-SSMActivation -DefaultInstanceName "MyWebServers" -IamRole
 "SSMAutomationRole" -RegistrationLimit 10
```

Sortie :

```
ActivationCode       ActivationId
--------------       ------------
KWChhOxBTiwDcKE9BlKC 08e51e79-1e36-446c-8e63-9458569c1363
```

- Pour plus de détails sur l'API, reportez-vous CreateActivationà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`CreateAssociation`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`CreateAssociation`.

CLI

AWS CLI

Exemple 1 : pour associer un document à l'aide des ID d'instance

Cet exemple associe un document de configuration à une instance à l'aide des ID d'instance.

```
aws ssm create-association \
    --instance-id "i-0cb2b964d3e14fd9f" \
    --name "AWS-UpdateSSMAgent"
```

Sortie :

```
{
    "AssociationDescription": {
        "Status": {
            "Date": 1487875500.33,
            "Message": "Associated with AWS-UpdateSSMAgent",
            "Name": "Associated"
        },
```

```
        "Name": "AWS-UpdateSSMAgent",
        "InstanceId": "i-0cb2b964d3e14fd9f",
        "Overview": {
            "Status": "Pending",
            "DetailedStatus": "Creating"
        },
        "AssociationId": "b7c3266e-a544-44db-877e-b20d3a108189",
        "DocumentVersion": "$DEFAULT",
        "LastUpdateAssociationDate": 1487875500.33,
        "Date": 1487875500.33,
        "Targets": [
            {
                "Values": [
                    "i-0cb2b964d3e14fd9f"
                ],
                "Key": "InstanceIds"
            }
        ]
    }
}
```

Pour plus d'informations, consultez CreateAssociationla référence de l'API AWS Systems Manager.

Exemple 2 : pour associer un document à l'aide de cibles

Cet exemple associe un document de configuration à une instance, en utilisant des cibles.

```
aws ssm create-association \
    --name "AWS-UpdateSSMAgent" \
    --targets "Key=instanceids,Values=i-0cb2b964d3e14fd9f"
```

Sortie :

```
{
    "AssociationDescription": {
        "Status": {
            "Date": 1487875500.33,
            "Message": "Associated with AWS-UpdateSSMAgent",
            "Name": "Associated"
        },
        "Name": "AWS-UpdateSSMAgent",
        "InstanceId": "i-0cb2b964d3e14fd9f",
```

```
        "Overview": {
            "Status": "Pending",
            "DetailedStatus": "Creating"
        },
        "AssociationId": "b7c3266e-a544-44db-877e-b20d3a108189",
        "DocumentVersion": "$DEFAULT",
        "LastUpdateAssociationDate": 1487875500.33,
        "Date": 1487875500.33,
        "Targets": [
            {
                "Values": [
                    "i-0cb2b964d3e14fd9f"
                ],
                "Key": "InstanceIds"
            }
        ]
    }
}
```

Pour plus d'informations, consultez CreateAssociationla référence de l'API AWS Systems Manager.

Exemple 3 : pour créer une association qui ne s'exécute qu'une seule fois

Cet exemple crée une nouvelle association qui ne s'exécute qu'une seule fois à la date et à l'heure spécifiées. Les associations créées avec une date passée ou présente (au moment où elles sont traitées, la date est passée) s'exécutent immédiatement.

```
aws ssm create-association \
    --name "AWS-UpdateSSMAgent" \
    --targets "Key=instanceids,Values=i-0cb2b964d3e14fd9f" \
    --schedule-expression "at(2020-05-14T15:55:00)"  \
    --apply-only-at-cron-interval
```

Sortie :

```
{
    "AssociationDescription": {
        "Status": {
            "Date": 1487875500.33,
            "Message": "Associated with AWS-UpdateSSMAgent",
            "Name": "Associated"
```

```
        },
        "Name": "AWS-UpdateSSMAgent",
        "InstanceId": "i-0cb2b964d3e14fd9f",
        "Overview": {
            "Status": "Pending",
            "DetailedStatus": "Creating"
        },
        "AssociationId": "b7c3266e-a544-44db-877e-b20d3a108189",
        "DocumentVersion": "$DEFAULT",
        "LastUpdateAssociationDate": 1487875500.33,
        "Date": 1487875500.33,
        "Targets": [
            {
                "Values": [
                    "i-0cb2b964d3e14fd9f"
                ],
                "Key": "InstanceIds"
            }
        ]
    }
}
```

Pour plus d'informations, reportez-vous CreateAssociationà la section Référence de l'API AWS Systems Manager ou Reference : Cron and rate expressions for Systems Manager dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous CreateAssociationà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple associe un document de configuration à une instance, à l'aide des ID d'instance.

```
New-SSMAssociation -InstanceId "i-0cb2b964d3e14fd9f" -Name "AWS-UpdateSSMAgent"
```

Sortie :

```
Name                 : AWS-UpdateSSMAgent
InstanceId           : i-0000293ffd8c57862
```

```
Date                  : 2/23/2017 6:55:22 PM
Status.Name           : Associated
Status.Date           : 2/20/2015 8:31:11 AM
Status.Message        : Associated with AWS-UpdateSSMAgent
Status.AdditionalInfo :
```

Exemple 2 : Cet exemple associe un document de configuration à une instance, en utilisant des cibles.

```
$target = @{Key="instanceids";Values=@("i-0cb2b964d3e14fd9f")}
New-SSMAssociation -Name "AWS-UpdateSSMAgent" -Target $target
```

Sortie :

```
Name                  : AWS-UpdateSSMAgent
InstanceId            :
Date                  : 3/1/2017 6:22:21 PM
Status.Name           :
Status.Date           :
Status.Message        :
Status.AdditionalInfo :
```

Exemple 3 : Cet exemple associe un document de configuration à une instance, à l'aide de cibles et de paramètres.

```
$target = @{Key="instanceids";Values=@("i-0cb2b964d3e14fd9f")}
$params = @{
  "action"="configure"
  "mode"="ec2"
  "optionalConfigurationSource"="ssm"
  "optionalConfigurationLocation"=""
  "optionalRestart"="yes"
}
New-SSMAssociation -Name "Configure-CloudWatch" -AssociationName
 "CWConfiguration" -Target $target -Parameter $params
```

Sortie :

```
Name                  : Configure-CloudWatch
InstanceId            :
Date                  : 5/17/2018 3:17:44 PM
```

```
Status.Name           :
Status.Date           :
Status.Message        :
Status.AdditionalInfo :
```

Exemple 4 : Cet exemple crée une association avec toutes les instances de la région, avec**`AWS-GatherSoftwareInventory`**. Il fournit également des fichiers personnalisés et des emplacements de registre dans les paramètres à collecter

```
$params =
 [Collections.Generic.Dictionary[String,Collections.Generic.List[String]]]::new()
$params["windowsRegistry"] ='[{"Path":"HKEY_LOCAL_MACHINE\SOFTWARE\Amazon
\MachineImage","Recursive":false,"ValueNames":["AMIName"]}]'
$params["files"] = '[{"Path":"C:\Program Files","Pattern":
["*.exe"],"Recursive":true}, {"Path":"C:\ProgramData","Pattern":
["*.log"],"Recursive":true}]'
New-SSMAssociation -AssociationName new-in-mum -Name AWS-GatherSoftwareInventory
 -Target @{Key="instanceids";Values="*"} -Parameter $params -region ap-south-1 -
ScheduleExpression "rate(720 minutes)"
```

Sortie :

```
Name                  : AWS-GatherSoftwareInventory
InstanceId            :
Date                  : 6/9/2019 8:57:56 AM
Status.Name           :
Status.Date           :
Status.Message        :
Status.AdditionalInfo :
```

- Pour plus de détails sur l'API, reportez-vous CreateAssociationà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`CreateAssociationBatch`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`CreateAssociationBatch`.

CLI

AWS CLI

Pour créer plusieurs associations

Cet exemple associe un document de configuration à plusieurs instances. La sortie renvoie une liste des opérations réussies et échouées, le cas échéant.

Commande :

```
aws ssm create-association-batch --entries "Name=AWS-
UpdateSSMAgent,InstanceId=i-1234567890abcdef0" "Name=AWS-
UpdateSSMAgent,InstanceId=i-9876543210abcdef0"
```

Sortie :

```
{
  "Successful": [
      {
          "Name": "AWS-UpdateSSMAgent",
          "InstanceId": "i-1234567890abcdef0",
          "AssociationVersion": "1",
          "Date": 1550504725.007,
          "LastUpdateAssociationDate": 1550504725.007,
          "Status": {
              "Date": 1550504725.007,
              "Name": "Associated",
              "Message": "Associated with AWS-UpdateSSMAgent"
          },
          "Overview": {
              "Status": "Pending",
              "DetailedStatus": "Creating"
          },
          "DocumentVersion": "$DEFAULT",
          "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
          "Targets": [
              {
                  "Key": "InstanceIds",
                  "Values": [
                      "i-1234567890abcdef0"
                  ]
              }
```

```
            ]
        },
        {
            "Name": "AWS-UpdateSSMAgent",
            "InstanceId": "i-9876543210abcdef0",
            "AssociationVersion": "1",
            "Date": 1550504725.057,
            "LastUpdateAssociationDate": 1550504725.057,
            "Status": {
                "Date": 1550504725.057,
                "Name": "Associated",
                "Message": "Associated with AWS-UpdateSSMAgent"
            },
            "Overview": {
                "Status": "Pending",
                "DetailedStatus": "Creating"
            },
            "DocumentVersion": "$DEFAULT",
            "AssociationId": "9c9f7f20-5154-4fed-a83e-0123456789ab",
            "Targets": [
                {
                    "Key": "InstanceIds",
                    "Values": [
                        "i-9876543210abcdef0"
                    ]
                }
            ]
        }
    ],
    "Failed": []
}
```

- Pour plus de détails sur l'API, consultez CreateAssociationBatch in AWS CLI Command Reference.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple associe un document de configuration à plusieurs instances. La sortie renvoie une liste des opérations réussies et échouées, le cas échéant.

```
$option1 = @{InstanceId="i-0cb2b964d3e14fd9f";Name=@("AWS-UpdateSSMAgent")}
$option2 = @{InstanceId="i-0000293ffd8c57862";Name=@("AWS-UpdateSSMAgent")}
New-SSMAssociationFromBatch -Entry $option1,$option2
```

Sortie :

```
Failed  Successful
------  ----------
{}      {Amazon.SimpleSystemsManagement.Model.FailedCreateAssociation,
 Amazon.SimpleSystemsManagement.Model.FailedCreateAsso...
```

Exemple 2 : Cet exemple montre tous les détails d'une opération réussie.

```
$option1 = @{InstanceId="i-0cb2b964d3e14fd9f";Name=@("AWS-UpdateSSMAgent")}
$option2 = @{InstanceId="i-0000293ffd8c57862";Name=@("AWS-UpdateSSMAgent")}
(New-SSMAssociationFromBatch -Entry $option1,$option2).Successful
```

- Pour plus de détails sur l'API, consultez CreateAssociationBatch in AWS Tools for PowerShell Cmdlet Reference.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`CreateDocument`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`CreateDocument`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- Commencez avec Systems Manager

CLI

AWS CLI

Pour créer un document

L'`create-document`exemple suivant crée un document Systems Manager.

```
aws ssm create-document \
    --content file://exampleDocument.yml \
    --name "Example" \
    --document-type "Automation" \
    --document-format YAML
```

Sortie :

```
{
    "DocumentDescription": {
        "Hash":
 "fc2410281f40779e694a8b95975d0f9f316da8a153daa94e3d9921102EXAMPLE",
        "HashType": "Sha256",
        "Name": "Example",
        "Owner": "29884EXAMPLE",
        "CreatedDate": 1583256349.452,
        "Status": "Creating",
        "DocumentVersion": "1",
        "Description": "Document Example",
        "Parameters": [
            {
                "Name": "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.",
                "DefaultValue": ""
            },
            {
                "Name": "InstanceId",
                "Type": "String",
                "Description": "(Required) The ID of the Amazon EC2 instance.",
                "DefaultValue": ""
            }
        ],
        "PlatformTypes": [
            "Windows",
            "Linux"
        ],
        "DocumentType": "Automation",
        "SchemaVersion": "0.3",
```

```
        "LatestVersion": "1",
        "DefaultVersion": "1",
        "DocumentFormat": "YAML",
        "Tags": []
    }
}
```

Pour plus d'informations, consultez la section Création de documents Systems Manager dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous CreateDocumentà la section Référence des AWS CLI commandes.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
    // Create an AWS SSM document to use in this scenario.
    public static void createSSMDoc(SsmClient ssmClient, String docName) {
        // Create JSON for the content
        String jsonData = """
            {
                "schemaVersion": "2.2",
                "description": "Run a simple shell command",
                "mainSteps": [
                    {
                        "action": "aws:runShellScript",
                        "name": "runEchoCommand",
                        "inputs": {
                          "runCommand": [
                            "echo 'Hello, world!'"
                          ]
                        }
                      }
                    ]
                }
```

```
            """;

        try {
            CreateDocumentRequest request = CreateDocumentRequest.builder()
                .content(jsonData)
                .name(docName)
                .documentType(DocumentType.COMMAND)
                .build();

            // Create the document.
            CreateDocumentResponse response = ssmClient.createDocument(request);
            System.out.println("The status of the document is " +
 response.documentDescription().status());

        } catch (DocumentAlreadyExistsException e) {
            System.err.println("The document already exists. Moving on." );
        } catch (SsmException e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }
```

- Pour plus de détails sur l'API, reportez-vous CreateDocumentà la section Référence des AWS SDK for Java 2.x API.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple crée un document dans votre compte. Le document doit être au format JSON. Pour plus d'informations sur la rédaction d'un document de configuration, consultez le document de configuration dans le manuel de référence de l'API SSM.

```
New-SSMDocument -Content (Get-Content -Raw "c:\temp\RunShellScript.json") -Name
 "RunShellScript" -DocumentType "Command"
```

Sortie :

```
CreatedDate     : 3/1/2017 1:21:33 AM
DefaultVersion  : 1
Description     : Run an updated script
```

```
DocumentType    : Command
DocumentVersion : 1
Hash            :
 1d5ce820e999ff051eb4841ed887593daf77120fd76cae0d18a53cc42e4e22c1
HashType        : Sha256
LatestVersion   : 1
Name            : RunShellScript
Owner           : 809632081692
Parameters      : {commands}
PlatformTypes   : {Linux}
SchemaVersion   : 2.0
Sha1            :
Status          : Creating
```

- Pour plus de détails sur l'API, reportez-vous CreateDocumentà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`CreateMaintenanceWindow`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`CreateMaintenanceWindow`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- Commencez avec Systems Manager

CLI

AWS CLI

Exemple 1 : pour créer une fenêtre de maintenance

L'`create-maintenance-window`exemple suivant crée une nouvelle fenêtre de maintenance qui, toutes les cinq minutes pendant deux heures au maximum (selon les besoins), empêche le démarrage de nouvelles tâches dans l'heure qui suit la fin de l'exécution de la fenêtre de

maintenance, autorise les cibles non associées (instances que vous n'avez pas enregistrées dans la fenêtre de maintenance) et indique par le biais de balises personnalisées que son créateur a l'intention de l'utiliser dans un didacticiel.

```
aws ssm create-maintenance-window \
    --name "My-Tutorial-Maintenance-Window" \
    --schedule "rate(5 minutes)" \
    --duration 2 --cutoff 1 \
    --allow-unassociated-targets \
    --tags "Key=Purpose,Value=Tutorial"
```

Sortie :

```
{
    "WindowId": "mw-0c50858d01EXAMPLE"
}
```

Exemple 2 : pour créer une fenêtre de maintenance qui ne s'exécute qu'une seule fois

L'`create-maintenance-window`exemple suivant crée une nouvelle fenêtre de maintenance qui ne s'exécute qu'une seule fois à la date et à l'heure spécifiées.

```
aws ssm create-maintenance-window \
    --name My-One-Time-Maintenance-Window \
    --schedule "at(2020-05-14T15:55:00)" \
    --duration 5 \
    --cutoff 2 \
    --allow-unassociated-targets \
    --tags "Key=Environment,Value=Production"
```

Sortie :

```
{
    "WindowId": "mw-01234567890abcdef"
}
```

Pour plus d'informations, consultez la section Maintenance Windows dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, voir CreateMaintenanceWindow in AWS CLI Command Reference.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
   public static String createMaintenanceWindow(SsmClient ssmClient, String
winName) {
       CreateMaintenanceWindowRequest request =
CreateMaintenanceWindowRequest.builder()
           .name(winName)
           .description("This is my maintenance window")
           .allowUnassociatedTargets(true)
           .duration(2)
           .cutoff(1)
           .schedule("cron(0 10 ? * MON-FRI *)")
           .build();

       try {
           CreateMaintenanceWindowResponse response =
ssmClient.createMaintenanceWindow(request);
           String maintenanceWindowId = response.windowId();
           System.out.println("The maintenance window id is " +
maintenanceWindowId);
           return maintenanceWindowId;

       } catch (DocumentAlreadyExistsException e) {
           System.err.println("The maintenance window already exists. Moving
on.");
       } catch (SsmException e) {
           System.err.println(e.getMessage());
           System.exit(1);
       }

       MaintenanceWindowFilter filter = MaintenanceWindowFilter.builder()
           .key("name")
           .values(winName)
           .build();
```

```
        DescribeMaintenanceWindowsRequest winRequest =
DescribeMaintenanceWindowsRequest.builder()
            .filters(filter)
            .build();

        String windowId = "";
        DescribeMaintenanceWindowsResponse response =
ssmClient.describeMaintenanceWindows(winRequest);
        List<MaintenanceWindowIdentity> windows = response.windowIdentities();
        if (!windows.isEmpty()) {
            windowId = windows.get(0).windowId();
            System.out.println("Window ID: " + windowId);
        } else {
            System.out.println("Window not found.");
        }
       return windowId;
   }
```

- Pour plus de détails sur l'API, voir CreateMaintenanceWindow in AWS SDK for Java 2.x API Reference.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple crée une nouvelle fenêtre de maintenance portant le nom spécifié qui s'exécute à 16 heures tous les mardis pendant 4 heures, avec une limite d'une heure, et qui autorise des cibles non associées.

```
New-SSMMaintenanceWindow -Name "MyMaintenanceWindow" -Duration 4 -Cutoff 1 -
AllowUnassociatedTarget $true -Schedule "cron(0 16 ? * TUE *)"
```

Sortie :

```
mw-03eb53e1ea7383998
```

- Pour plus de détails sur l'API, consultez CreateMaintenancela section Window in AWS Tools for PowerShell Cmdlet Reference.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`CreateOpsItem`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`CreateOpsItem`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- Commencez avec Systems Manager

CLI

AWS CLI

Pour créer un OpsItems

L'`create-ops-item`exemple suivant utilise la clé /aws/resources OperationalData pour créer un OpsItem avec une ressource associée à Amazon DynamoDB.

```
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"
```

Sortie :

```
{
    "OpsItemId": "oi-1a2b3c4d5e6f"
}
```

Pour plus d'informations, voir Creating OpsItems in the AWS Systems Manager User Guide.

- Pour plus de détails sur l'API, voir CreateOpsÉlément dans AWS CLI la référence des commandes.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
   // Create an SSM OpsItem
   public static String createSSMOpsItem(SsmClient ssmClient, String title,
String source, String category, String severity) {
       try {
           CreateOpsItemRequest opsItemRequest = CreateOpsItemRequest.builder()
               .description("Created by the Systems Manager Java API")
               .title(title)
               .source(source)
               .category(category)
               .severity(severity)
               .build();

           CreateOpsItemResponse itemResponse =
ssmClient.createOpsItem(opsItemRequest);
           return itemResponse.opsItemId();

       } catch (SsmException e) {
           System.err.println(e.getMessage());
           System.exit(1);
       }
      return "";
   }
```

- Pour plus de détails sur l'API, voir CreateOpsl'élément dans le guide de référence des AWS SDK for Java 2.x API.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`CreatePatchBaseline`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`CreatePatchBaseline`.

CLI

AWS CLI

Exemple 1 : pour créer une référence de correctifs avec approbation automatique

L'`create-patch-baseline`exemple suivant crée une ligne de base de correctifs pour Windows Server qui approuve les correctifs pour un environnement de production sept jours après leur publication par Microsoft.

```
aws ssm create-patch-baseline \
    --name "Windows-Production-Baseline-AutoApproval" \
    --operating-system "WINDOWS" \
    --approval-rules
 "PatchRules=[{PatchFilterGroup={PatchFilters=[{Key=MSRC_SEVERITY,Values=[Critical,Import
{Key=CLASSIFICATION,Values=[SecurityUpdates,Updates,UpdateRollups,CriticalUpdates]}]},App
 \
    --description "Baseline containing all updates approved for Windows Server
 production systems"
```

Sortie :

```
{
    "BaselineId": "pb-045f10b4f3EXAMPLE"
}
```

Exemple 2 : pour créer une référence de correctifs avec une date limite d'approbation

L'`create-patch-baseline`exemple suivant crée une ligne de base de correctifs pour Windows Server qui approuve tous les correctifs d'un environnement de production publiés le 7 juillet 2020 ou avant.

```
aws ssm create-patch-baseline \
    --name "Windows-Production-Baseline-AutoApproval" \
```

```
    --operating-system "WINDOWS" \
    --approval-rules
 "PatchRules=[{PatchFilterGroup={PatchFilters=[{Key=MSRC_SEVERITY,Values=[Critical,Import
{Key=CLASSIFICATION,Values=[SecurityUpdates,Updates,UpdateRollups,CriticalUpdates]}]},App
 \
    --description "Baseline containing all updates approved for Windows Server
 production systems"
```

Sortie :

```
{
    "BaselineId": "pb-045f10b4f3EXAMPLE"
}
```

Exemple 3 : pour créer une ligne de base de correctifs avec des règles d'approbation stockées dans un fichier JSON

L'`create-patch-baseline`exemple suivant crée une ligne de base de correctifs pour Amazon Linux 2017.09 qui approuve les correctifs pour un environnement de production sept jours après leur publication, spécifie les règles d'approbation pour la ligne de base de correctifs et spécifie un référentiel personnalisé pour les correctifs.

```
aws ssm create-patch-baseline \
    --cli-input-json file://my-amazon-linux-approval-rules-and-repo.json
```

Contenu de `my-amazon-linux-approval-rules-and-repo.json` :

```
{
    "Name": "Amazon-Linux-2017.09-Production-Baseline",
    "Description": "My approval rules patch baseline for Amazon Linux 2017.09
 instances",
    "OperatingSystem": "AMAZON_LINUX",
    "Tags": [
        {
            "Key": "Environment",
            "Value": "Production"
        }
    ],
    "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"
        }
    ]
}
```

Exemple 4 : pour créer une ligne de base de correctifs qui spécifie les correctifs approuvés et rejetés

L'`create-patch-baseline`exemple suivant indique explicitement les correctifs à approuver et à rejeter en tant qu'exception aux règles d'approbation par défaut.

```
aws ssm create-patch-baseline \
    --name "Amazon-Linux-2017.09-Alpha-Baseline" \
    --description "My custom approve/reject patch baseline for Amazon Linux
 2017.09 instances" \
    --operating-system "AMAZON_LINUX" \
    --approved-patches "CVE-2018-1234567,example-pkg-EE-2018*.amzn1.noarch" \
    --approved-patches-compliance-level "HIGH" \
    --approved-patches-enable-non-security \
    --tags "Key=Environment,Value=Alpha"
```

Pour plus d'informations, voir Create a Custom Patch Baseline dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, voir CreatePatchBaseline dans AWS CLI Command Reference.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple crée une ligne de base de correctifs qui approuve les correctifs, sept jours après leur publication par Microsoft, pour les instances gérées exécutant Windows Server 2019 dans un environnement de production.

```
$rule = New-Object Amazon.SimpleSystemsManagement.Model.PatchRule
$rule.ApproveAfterDays = 7

$ruleFilters = New-Object Amazon.SimpleSystemsManagement.Model.PatchFilterGroup

$patchFilter = New-Object Amazon.SimpleSystemsManagement.Model.PatchFilter
$patchFilter.Key="PRODUCT"
$patchFilter.Values="WindowsServer2019"

$severityFilter = New-Object Amazon.SimpleSystemsManagement.Model.PatchFilter
$severityFilter.Key="MSRC_SEVERITY"
$severityFilter.Values.Add("Critical")
```

```
$severityFilter.Values.Add("Important")
$severityFilter.Values.Add("Moderate")

$classificationFilter = New-Object
 Amazon.SimpleSystemsManagement.Model.PatchFilter
$classificationFilter.Key = "CLASSIFICATION"
$classificationFilter.Values.Add( "SecurityUpdates" )
$classificationFilter.Values.Add( "Updates" )
$classificationFilter.Values.Add( "UpdateRollups" )
$classificationFilter.Values.Add( "CriticalUpdates" )

$ruleFilters.PatchFilters.Add($severityFilter)
$ruleFilters.PatchFilters.Add($classificationFilter)
$ruleFilters.PatchFilters.Add($patchFilter)
$rule.PatchFilterGroup = $ruleFilters

New-SSMPatchBaseline -Name "Production-Baseline-Windows2019" -Description
 "Baseline containing all updates approved for production systems" -
ApprovalRules_PatchRule $rule
```

Sortie :

```
pb-0z4z6221c4296b23z
```

- Pour plus de détails sur l'API, consultez la section CreatePatchBaseline dans la référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DeleteActivation`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DeleteActivation`.

CLI

AWS CLI

Pour supprimer l'activation d'une instance gérée

L'`delete-activation`exemple suivant supprime l'activation d'une instance gérée.

```
aws ssm delete-activation \
    --activation-id "aa673477-d926-42c1-8757-1358cEXAMPLE"
```

Cette commande ne produit aucun résultat.

Pour plus d'informations, consultez la section Configuration de AWS Systems Manager pour les environnements hybrides dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous DeleteActivationà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple supprime une activation. Il n'y a aucune sortie si la commande aboutit.

```
Remove-SSMActivation -ActivationId "08e51e79-1e36-446c-8e63-9458569c1363"
```

- Pour plus de détails sur l'API, reportez-vous DeleteActivationà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DeleteAssociation`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DeleteAssociation`.

CLI

AWS CLI

Exemple 1 : pour supprimer une association à l'aide de l'ID d'association

L'`delete-association`exemple suivant supprime l'association pour l'ID d'association spécifié. Il n'y a pas de sortie si la commande réussit.

```
aws ssm delete-association \
```

```
    --association-id "8dfe3659-4309-493a-8755-0123456789ab"
```

Cette commande ne produit aucun résultat.

Pour plus d'informations, reportez-vous à la section Modification et création d'une nouvelle version d'une association dans le Guide de l'utilisateur de AWS Systems Manager.

Exemple 2 : pour supprimer une association

L'`delete-association`exemple suivant supprime l'association entre une instance et un document. Il n'y a pas de sortie si la commande réussit.

```
aws ssm delete-association \
    --instance-id "i-1234567890abcdef0" \
    --name "AWS-UpdateSSMAgent"
```

Cette commande ne produit aucun résultat.

Pour plus d'informations, reportez-vous à la section Utilisation des associations dans Systems Manager dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous DeleteAssociationà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple supprime l'association entre une instance et un document. Il n'y a aucune sortie si la commande aboutit.

```
Remove-SSMAssociation -InstanceId "i-0cb2b964d3e14fd9f" -Name "AWS-
UpdateSSMAgent"
```

- Pour plus de détails sur l'API, reportez-vous DeleteAssociationà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DeleteDocument`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DeleteDocument`.

CLI

AWS CLI

Pour supprimer un document

L'`delete-document`exemple suivant supprime un document Systems Manager.

```
aws ssm delete-document \
    --name "Example"
```

Cette commande ne produit aucun résultat.

Pour plus d'informations, consultez la section Création de documents Systems Manager dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous DeleteDocumentà la section Référence des AWS CLI commandes.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
    // Deletes an AWS Systems Manager document.
    public static void deleteDoc(SsmClient ssmClient, String documentName) {
        try {
            DeleteDocumentRequest documentRequest =
DeleteDocumentRequest.builder()
                .name(documentName)
                .build();
```

```
            ssmClient.deleteDocument(documentRequest);
            System.out.println("The Systems Manager document was successfully
deleted.");

        } catch (SsmException e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }
```

- Pour plus de détails sur l'API, reportez-vous DeleteDocumentà la section Référence des AWS SDK for Java 2.x API.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple supprime un document. Il n'y a aucune sortie si la commande aboutit.

```
Remove-SSMDocument -Name "RunShellScript"
```

- Pour plus de détails sur l'API, reportez-vous DeleteDocumentà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DeleteMaintenanceWindow`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DeleteMaintenanceWindow`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- Commencez avec Systems Manager

CLI

AWS CLI

Pour supprimer une fenêtre de maintenance

Cet `delete-maintenance-window` exemple supprime la fenêtre de maintenance spécifiée.

```
aws ssm delete-maintenance-window \
    --window-id "mw-1a2b3c4d5e6f7g8h9"
```

Sortie :

```
{
    "WindowId":"mw-1a2b3c4d5e6f7g8h9"
}
```

Pour plus d'informations, voir Supprimer une fenêtre de maintenance (AWS CLI) dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, voir DeleteMaintenanceWindow in AWS CLI Command Reference.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
   public static void deleteMaintenanceWindow(SsmClient ssmClient, String winId)
{
       try {
           DeleteMaintenanceWindowRequest windowRequest =
DeleteMaintenanceWindowRequest.builder()
               .windowId(winId)
               .build();
```

```
            ssmClient.deleteMaintenanceWindow(windowRequest);
            System.out.println("The maintenance window was successfully
deleted.");

        } catch (SsmException e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }
```

- Pour plus de détails sur l'API, voir DeleteMaintenanceWindow in AWS SDK for Java 2.x API Reference.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple supprime une fenêtre de maintenance.

```
Remove-SSMMaintenanceWindow -WindowId "mw-06d59c1a07c022145"
```

Sortie :

```
mw-06d59c1a07c022145
```

- Pour plus de détails sur l'API, consultez DeleteMaintenancela section Window in AWS Tools for PowerShell Cmdlet Reference.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DeleteParameter`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DeleteParameter`.

CLI

AWS CLI

Pour supprimer un paramètre

L'`delete-parameter`exemple suivant supprime le paramètre unique spécifié.

```
aws ssm delete-parameter \
    --name "MyParameter"
```

Cette commande ne produit aucun résultat.

Pour plus d'informations, consultez la section Working with Parameter Store dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, voir DeleteParameterla section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : cet exemple supprime un paramètre. Il n'y a aucune sortie si la commande aboutit.

```
Remove-SSMParameter -Name "helloWorld"
```

- Pour plus de détails sur l'API, consultez la section DeleteParameterRéférence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DeletePatchBaseline`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DeletePatchBaseline`.

CLI

AWS CLI

Pour supprimer une ligne de base de correctif

L'`delete-patch-baseline`exemple suivant supprime la ligne de base de correctif spécifiée.

```
aws ssm delete-patch-baseline \
    --baseline-id "pb-045f10b4f382baeda"
```

Sortie :

```
{
    "BaselineId": "pb-045f10b4f382baeda"
}
```

Pour plus d'informations, consultez la section Mettre à jour ou supprimer une ligne de base de correctifs (console) dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, voir DeletePatchBaseline dans AWS CLI Command Reference.

PowerShell

Outils pour PowerShell

Exemple 1 : cet exemple supprime une ligne de base de correctif.

```
Remove-SSMPatchBaseline -BaselineId "pb-045f10b4f382baeda"
```

Sortie :

```
pb-045f10b4f382baeda
```

- Pour plus de détails sur l'API, consultez la section DeletePatchBaseline dans la référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DeregisterManagedInstance`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DeregisterManagedInstance`.

CLI

AWS CLI

Pour désenregistrer une instance gérée

L'`deregister-managed-instance`exemple suivant annule l'enregistrement de l'instance gérée spécifiée.

```
aws ssm deregister-managed-instance
    --instance-id "mi-08ab247cdfEXAMPLE"
```

Cette commande ne produit aucun résultat.

Pour plus d'informations, consultez la section Désenregistrement des instances gérées dans un environnement hybride dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, consultez DeregisterManagedInstance in AWS CLI Command Reference.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple annule l'enregistrement d'une instance gérée. Il n'y a aucune sortie si la commande aboutit.

```
Unregister-SSMManagedInstance -InstanceId "mi-08ab247cdf1046573"
```

- Pour plus de détails sur l'API, consultez la section DeregisterManagedInstance in AWS Tools for PowerShell Cmdlet Reference.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **DeregisterPatchBaselineForPatchGroup** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DeregisterPatchBaselineForPatchGroup`.

CLI

AWS CLI

Pour désenregistrer un groupe de correctifs depuis une ligne de base de correctifs

L'`deregister-patch-baseline-for-patch-group`exemple suivant permet de désenregistrer le groupe de correctifs spécifié de la ligne de base de correctifs spécifiée.

```
aws ssm deregister-patch-baseline-for-patch-group \
    --patch-group "Production" \
    --baseline-id "pb-0ca44a362fEXAMPLE"
```

Sortie :

```
{
  "PatchGroup":"Production",
  "BaselineId":"pb-0ca44a362fEXAMPLE"
}
```

Pour plus d'informations, consultez la section Ajouter un groupe de correctifs à une ligne de base de correctifs dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous DeregisterPatchBaselineForPatchGroupà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : cet exemple désenregistre un groupe de correctifs d'une ligne de base de correctifs.

```
Unregister-SSMPatchBaselineForPatchGroup -BaselineId "pb-045f10b4f382baeda" -
PatchGroup "Production"
```

Sortie :

```
BaselineId           PatchGroup
----------           ----------
pb-045f10b4f382baeda Production
```

- Pour plus de détails sur l'API, reportez-vous DeregisterPatchBaselineForPatchGroupà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DeregisterTargetFromMaintenanceWindow`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DeregisterTargetFromMaintenanceWindow`.

CLI

AWS CLI

Pour supprimer une cible d'une fenêtre de maintenance

L'`deregister-target-from-maintenance-window`exemple suivant supprime la cible spécifiée de la fenêtre de maintenance spécifiée.

```
aws ssm deregister-target-from-maintenance-window \
```

```
    --window-id "mw-ab12cd34ef56gh78" \
    --window-target-id "1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
```

Sortie :

```
{
    "WindowId":"mw-ab12cd34ef56gh78",
    "WindowTargetId":"1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
}
```

Pour plus d'informations, voir Mettre à jour une fenêtre de maintenance (AWS CLI) dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, voir DeregisterTargetFromMaintenanceWindow in AWS CLI Command Reference.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple supprime une cible d'une fenêtre de maintenance.

```
Unregister-SSMTargetFromMaintenanceWindow -WindowTargetId
 "6ab5c208-9fc4-4697-84b7-b02a6cc25f7d" -WindowId "mw-06cf17cbefcb4bf4f"
```

Sortie :

```
WindowId             WindowTargetId
--------             --------------
mw-06cf17cbefcb4bf4f 6ab5c208-9fc4-4697-84b7-b02a6cc25f7d
```

- Pour plus de détails sur l'API, consultez DeregisterTargetFromMaintenancela section Window in AWS Tools for PowerShell Cmdlet Reference.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **DeregisterTaskFromMaintenanceWindow** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DeregisterTaskFromMaintenanceWindow`.

CLI

AWS CLI

Pour supprimer une tâche d'une fenêtre de maintenance

L'`deregister-task-from-maintenance-window`exemple suivant supprime la tâche spécifiée de la fenêtre de maintenance spécifiée.

```
aws ssm deregister-task-from-maintenance-window \
    --window-id "mw-ab12cd34ef56gh78" \
    --window-task-id "1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d5e6c"
```

Sortie :

```
{
    "WindowTaskId":"1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d5e6c",
    "WindowId":"mw-ab12cd34ef56gh78"
}
```

Pour plus d'informations, consultez les didacticiels Windows (AWS CLI) de maintenance de Systems Manager dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, voir DeregisterTaskFromMaintenanceWindow in AWS CLI Command Reference.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple supprime une tâche d'une fenêtre de maintenance.

```
Unregister-SSMTaskFromMaintenanceWindow -WindowTaskId "f34a2c47-ddfd-4c85-
a88d-72366b69af1b" -WindowId "mw-03a342e62c96d31b0"
```

Sortie :

```
WindowId             WindowTaskId
--------             ------------
mw-03a342e62c96d31b0 f34a2c47-ddfd-4c85-a88d-72366b69af1b
```

- Pour plus de détails sur l'API, consultez DeregisterTaskFromMaintenancela section Window in AWS Tools for PowerShell Cmdlet Reference.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeActivations`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribeActivations`.

CLI

AWS CLI

Pour décrire les activations

L'`describe-activations`exemple suivant répertorie les informations relatives aux activations de votre AWS compte.

```
aws ssm describe-activations
```

Sortie :

```
{
    "ActivationList": [
        {
            "ActivationId": "5743558d-563b-4457-8682-d16c3EXAMPLE",
            "Description": "Example1",
            "IamRole": "HybridWebServersRole,
            "RegistrationLimit": 5,
            "RegistrationsCount": 5,
            "ExpirationDate": 1584316800.0,
            "Expired": false,
            "CreatedDate": 1581954699.792
```

```
        },
        {
            "ActivationId": "3ee0322b-f62d-40eb-b672-13ebfEXAMPLE",
            "Description": "Example2",
            "IamRole": "HybridDatabaseServersRole",
            "RegistrationLimit": 5,
            "RegistrationsCount": 5,
            "ExpirationDate": 1580515200.0,
            "Expired": true,
            "CreatedDate": 1578064132.002
        },
    ]
}
```

Pour plus d'informations, consultez Étape 4 : Création d'une activation d'instance gérée pour un environnement hybride dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous DescribeActivationsà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple fournit des détails sur les activations de votre compte.

```
Get-SSMActivation
```

Sortie :

```
ActivationId        : 08e51e79-1e36-446c-8e63-9458569c1363
CreatedDate         : 3/1/2017 12:01:51 AM
DefaultInstanceName : MyWebServers
Description         :
ExpirationDate      : 3/2/2017 12:01:51 AM
Expired             : False
IamRole             : AutomationRole
RegistrationLimit   : 10
RegistrationsCount  : 0
```

- Pour plus de détails sur l'API, reportez-vous DescribeActivationsà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeAssociation`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribeAssociation`.

CLI

AWS CLI

Exemple 1 : Pour obtenir les détails d'une association

L'`describe-association`exemple suivant décrit l'association pour l'ID d'association spécifié.

```
aws ssm describe-association \
    --association-id "8dfe3659-4309-493a-8755-0123456789ab"
```

Sortie :

```
{
    "AssociationDescription": {
        "Name": "AWS-GatherSoftwareInventory",
        "AssociationVersion": "1",
        "Date": 1534864780.995,
        "LastUpdateAssociationDate": 1543235759.81,
        "Overview": {
            "Status": "Success",
            "AssociationStatusAggregatedCount": {
                "Success": 2
            }
        },
        "DocumentVersion": "$DEFAULT",
        "Parameters": {
            "applications": [
                "Enabled"
            ],
            "awsComponents": [
                "Enabled"
            ],
```

```
            "customInventory": [
                "Enabled"
            ],
            "files": [
                ""
            ],
            "instanceDetailedInformation": [
                "Enabled"
            ],
            "networkConfig": [
                "Enabled"
            ],
            "services": [
                "Enabled"
            ],
            "windowsRegistry": [
                ""
            ],
            "windowsRoles": [
                "Enabled"
            ],
            "windowsUpdates": [
                "Enabled"
            ]
        },
        "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
        "Targets": [
            {
                "Key": "InstanceIds",
                "Values": [
                    "*"
                ]
            }
        ],
        "ScheduleExpression": "rate(24 hours)",
        "LastExecutionDate": 1550501886.0,
        "LastSuccessfulExecutionDate": 1550501886.0,
        "AssociationName": "Inventory-Association"
    }
}
```


Pour plus d'informations, reportez-vous à la section Modification et création d'une nouvelle version d'une association dans le Guide de l'utilisateur de AWS Systems Manager.

Exemple 2 : pour obtenir les détails d'une association pour une instance et un document spécifiques

L'`describe-association`exemple suivant décrit l'association entre une instance et un document.

```
aws ssm describe-association \
    --instance-id "i-1234567890abcdef0" \
    --name "AWS-UpdateSSMAgent"
```

Sortie :

```
{
    "AssociationDescription": {
        "Status": {
            "Date": 1487876122.564,
            "Message": "Associated with AWS-UpdateSSMAgent",
            "Name": "Associated"
        },
        "Name": "AWS-UpdateSSMAgent",
        "InstanceId": "i-1234567890abcdef0",
        "Overview": {
            "Status": "Pending",
            "DetailedStatus": "Associated",
            "AssociationStatusAggregatedCount": {
                "Pending": 1
            }
        },
        "AssociationId": "d8617c07-2079-4c18-9847-1234567890ab",
        "DocumentVersion": "$DEFAULT",
        "LastUpdateAssociationDate": 1487876122.564,
        "Date": 1487876122.564,
        "Targets": [
            {
                "Values": [
                    "i-1234567890abcdef0"
                ],
                "Key": "InstanceIds"
            }
        ]
    }
}
```

Pour plus d'informations, reportez-vous à la section Modification et création d'une nouvelle version d'une association dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous DescribeAssociationà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple décrit l'association entre une instance et un document.

```
Get-SSMAssociation -InstanceId "i-0000293ffd8c57862" -Name "AWS-UpdateSSMAgent"
```

Sortie :

```
Name                  : AWS-UpdateSSMAgent
InstanceId            : i-0000293ffd8c57862
Date                  : 2/23/2017 6:55:22 PM
Status.Name           : Pending
Status.Date           : 2/20/2015 8:31:11 AM
Status.Message        : temp_status_change
Status.AdditionalInfo : Additional-Config-Needed
```

- Pour plus de détails sur l'API, reportez-vous DescribeAssociationà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeAssociationExecutionTargets`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribeAssociationExecutionTargets`.

CLI

AWS CLI

Pour obtenir les détails de l'exécution d'une association

L'`describe-association-execution-targets`exemple suivant décrit l'exécution de l'association spécifiée.

```
aws ssm describe-association-execution-targets \
    --association-id "8dfe3659-4309-493a-8755-0123456789ab" \
    --execution-id "7abb6378-a4a5-4f10-8312-0123456789ab"
```

Sortie :

```
{
    "AssociationExecutionTargets": [
        {
            "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
            "AssociationVersion": "1",
            "ExecutionId": "7abb6378-a4a5-4f10-8312-0123456789ab",
            "ResourceId": "i-1234567890abcdef0",
            "ResourceType": "ManagedInstance",
            "Status": "Success",
            "DetailedStatus": "Success",
            "LastExecutionDate": 1550505538.497,
            "OutputSource": {
                "OutputSourceId": "97fff367-fc5a-4299-aed8-0123456789ab",
                "OutputSourceType": "RunCommand"
            }
        }
    ]
}
```

Pour plus d'informations, reportez-vous à la section Affichage de l'historique des associations dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous DescribeAssociationExecutionTargetsà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple affiche l'ID de ressource et son statut d'exécution qui font partie des cibles d'exécution de l'association

```
Get-SSMAssociationExecutionTarget -AssociationId 123a45a0-
c678-9012-3456-78901234db5e -ExecutionId 123a45a0-c678-9012-3456-78901234db5e |
 Select-Object ResourceId, Status
```

Sortie :

```
ResourceId           Status
----------           ------
i-0b1b2a3456f7a890b  Success
i-01c12a45d6fc7a89f  Success
i-0a1caf234f56d7dc8  Success
i-012a3fd45af6dbcfe  Failed
i-0ddc1df23c4a5fb67  Success
```

Exemple 2 : Cette commande vérifie l'exécution particulière d'une automatisation particulière depuis hier, où un document de commande est associé. Il vérifie en outre si l'exécution de l'association a échoué, et si c'est le cas, il affichera les détails de l'appel de commande pour l'exécution ainsi que l'identifiant de l'instance

```
$AssociationExecution= Get-SSMAssociationExecutionTarget -
AssociationId 1c234567-890f-1aca-a234-5a678d901cb0 -ExecutionId
 12345ca12-3456-2345-2b45-23456789012 |
    Where-Object {$_.LastExecutionDate -gt (Get-Date -Hour 00 -Minute
 00).AddDays(-1)}

foreach ($execution in $AssociationExecution) {
    if($execution.Status -ne 'Success'){
        Write-Output "There was an issue executing the association
 $($execution.AssociationId) on $($execution.ResourceId)"
        Get-SSMCommandInvocation -CommandId
 $execution.OutputSource.OutputSourceId -Detail:$true | Select-Object -
ExpandProperty CommandPlugins
    }
}
```

Sortie :

```
There was an issue executing the association 1c234567-890f-1aca-a234-5a678d901cb0
 on i-0a1caf234f56d7dc8


Name                   : aws:runPowerShellScript
Output                 :
                         ----------ERROR-------
                         failed to run commands: exit status 1
OutputS3BucketName     :
OutputS3KeyPrefix      :
OutputS3Region         : eu-west-1
ResponseCode           : 1
ResponseFinishDateTime : 5/29/2019 11:04:49 AM
ResponseStartDateTime  : 5/29/2019 11:04:49 AM
StandardErrorUrl       :
StandardOutputUrl      :
Status                 : Failed
StatusDetails          : Failed
```

- Pour plus de détails sur l'API, reportez-vous DescribeAssociationExecutionTargetsà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeAssociationExecutions`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribeAssociationExecutions`.

CLI

AWS CLI

Exemple 1 : pour obtenir le détail de toutes les exécutions pour une association

L'`describe-association-executions`exemple suivant décrit toutes les exécutions de l'association spécifiée.

```
aws ssm describe-association-executions \
    --association-id "8dfe3659-4309-493a-8755-0123456789ab"
```

Sortie :

```
{
    "AssociationExecutions": [
        {
            "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
            "AssociationVersion": "1",
            "ExecutionId": "474925ef-1249-45a2-b93d-0123456789ab",
            "Status": "Success",
            "DetailedStatus": "Success",
            "CreatedTime": 1550505827.119,
            "ResourceCountByStatus": "{Success=1}"
        },
        {
            "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
            "AssociationVersion": "1",
            "ExecutionId": "7abb6378-a4a5-4f10-8312-0123456789ab",
            "Status": "Success",
            "DetailedStatus": "Success",
            "CreatedTime": 1550505536.843,
            "ResourceCountByStatus": "{Success=1}"
        },
        ...
    ]
}
```

Pour plus d'informations, reportez-vous à la section Affichage de l'historique des associations dans le Guide de l'utilisateur de AWS Systems Manager.

Exemple 2 : pour obtenir le détail de toutes les exécutions d'une association après une date et une heure spécifiques

L'`describe-association-executions`exemple suivant décrit toutes les exécutions d'une association après la date et l'heure spécifiées.

```
aws ssm describe-association-executions \
    --association-id "8dfe3659-4309-493a-8755-0123456789ab" \
    --filters "Key=CreatedTime,Value=2019-02-18T16:00:00Z,Type=GREATER_THAN"
```

Sortie :

```
{
    "AssociationExecutions": [
        {
            "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
            "AssociationVersion": "1",
            "ExecutionId": "474925ef-1249-45a2-b93d-0123456789ab",
            "Status": "Success",
            "DetailedStatus": "Success",
            "CreatedTime": 1550505827.119,
            "ResourceCountByStatus": "{Success=1}"
        },
        {
            "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
            "AssociationVersion": "1",
            "ExecutionId": "7abb6378-a4a5-4f10-8312-0123456789ab",
            "Status": "Success",
            "DetailedStatus": "Success",
            "CreatedTime": 1550505536.843,
            "ResourceCountByStatus": "{Success=1}"
        },
        ...
    ]
}
```

Pour plus d'informations, reportez-vous à la section Affichage de l'historique des associations dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, consultez la section DescribeAssociationExécutions dans AWS CLI la référence des commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple renvoie les exécutions pour l'ID d'association fourni

```
Get-SSMAssociationExecution -AssociationId 123a45a0-c678-9012-3456-78901234db5e
```

Sortie :

```
AssociationId         : 123a45a0-c678-9012-3456-78901234db5e
AssociationVersion    : 2
CreatedTime           : 3/2/2019 8:53:29 AM
DetailedStatus        :
ExecutionId           : 123a45a0-c678-9012-3456-78901234db5e
LastExecutionDate     : 1/1/0001 12:00:00 AM
ResourceCountByStatus : {Success=4}
Status                : Success
```

- Pour plus de détails sur l'API, consultez la section DescribeAssociationExecutions in AWS Tools for PowerShell Cmdlet Reference.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeAutomationExecutions`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribeAutomationExecutions`.

CLI

AWS CLI

Pour décrire une exécution d'automatisation

L'`describe-automation-executions`exemple suivant affiche les détails d'une exécution automatique.

```
aws ssm describe-automation-executions \
    --filters Key=ExecutionId,Values=73c8eef8-f4ee-4a05-820c-e354fEXAMPLE
```

Sortie :

```
{
    "AutomationExecutionMetadataList": [
        {
            "AutomationExecutionId": "73c8eef8-f4ee-4a05-820c-e354fEXAMPLE",
            "DocumentName": "AWS-StartEC2Instance",
```

```
            "DocumentVersion": "1",
            "AutomationExecutionStatus": "Success",
            "ExecutionStartTime": 1583737233.748,
            "ExecutionEndTime": 1583737234.719,
            "ExecutedBy": "arn:aws:sts::29884EXAMPLE:assumed-role/
mw_service_role/OrchestrationService",
            "LogFile": "",
            "Outputs": {},
            "Mode": "Auto",
            "Targets": [],
            "ResolvedTargets": {
                "ParameterValues": [],
                "Truncated": false
            },
            "AutomationType": "Local"
        }
    ]
}
```

Pour plus d'informations, consultez Running a Simple Automation Workflow dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, consultez la section DescribeAutomationExécutions dans AWS CLI la référence des commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple décrit toutes les exécutions automatisées actives et terminées associées à votre compte.

```
Get-SSMAutomationExecutionList
```

Sortie :

```
AutomationExecutionId     : 4105a4fc-f944-11e6-9d32-8fb2db27a909
AutomationExecutionStatus : Failed
DocumentName              : AWS-UpdateLinuxAmi
DocumentVersion           : 1
ExecutedBy                : admin
ExecutionEndTime          : 2/22/2017 9:17:08 PM
```

```
ExecutionStartTime        : 2/22/2017 9:17:02 PM
LogFile                   :
Outputs                   : {[createImage.ImageId,
 Amazon.Runtime.Internal.Util.AlwaysSendList`1[System.String]]}
```

Exemple 2 : Cet exemple affiche l'ExecutionID, le document, l'horodatage de début/fin de l'exécution pour les exécutions dont la valeur n'est pas « Success » AutomationExecutionStatus

```
Get-SSMAutomationExecutionList | Where-Object AutomationExecutionStatus
 -ne "Success" | Select-Object AutomationExecutionId, DocumentName,
 AutomationExecutionStatus, ExecutionStartTime, ExecutionEndTime | Format-Table -
AutoSize
```

Sortie :

```
AutomationExecutionId                DocumentName
 AutomationExecutionStatus ExecutionStartTime   ExecutionEndTime
--------------------                 ------------
 ------------------------ ------------------   ----------------
e1d2bad3-4567-8901-ae23-456c7c8901be AWS-UpdateWindowsAmi
 Cancelled                 4/16/2019 5:37:04 AM 4/16/2019 5:47:29 AM
61234567-a7f8-90e1-2b34-567b8bf9012c Fixed-UpdateAmi
 Cancelled                 4/16/2019 5:33:04 AM 4/16/2019 5:40:15 AM
91234d56-7e89-0ac1-2aee-34ea5d6a7c89 AWS-UpdateWindowsAmi
 Failed                    4/16/2019 5:22:46 AM 4/16/2019 5:27:29 AM
```

- Pour plus de détails sur l'API, consultez la section DescribeAutomationExecutions in AWS Tools for PowerShell Cmdlet Reference.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeAutomationStepExecutions`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribeAutomationStepExecutions`.

CLI

AWS CLI

Exemple 1 : Pour décrire toutes les étapes d'une exécution automatisée

L'`describe-automation-step-executions`exemple suivant affiche des détails sur les étapes d'une exécution automatique.

```
aws ssm describe-automation-step-executions \
    --automation-execution-id 73c8eef8-f4ee-4a05-820c-e354fEXAMPLE
```

Sortie :

```
{
    "StepExecutions": [
        {
            "StepName": "startInstances",
            "Action": "aws:changeInstanceState",
            "ExecutionStartTime": 1583737234.134,
            "ExecutionEndTime": 1583737234.672,
            "StepStatus": "Success",
            "Inputs": {
                "DesiredState": "\"running\"",
                "InstanceIds": "[\"i-0cb99161f6EXAMPLE\"]"
            },
            "Outputs": {
                "InstanceStates": [
                    "running"
                ]
            },
            "StepExecutionId": "95e70479-cf20-4d80-8018-7e4e2EXAMPLE",
            "OverriddenParameters": {}
        }
    ]
}
```

Exemple 2 : pour décrire une étape spécifique d'une exécution d'automatisation

L'`describe-automation-step-executions`exemple suivant affiche les détails d'une étape spécifique d'une exécution d'automatisation.

```
aws ssm describe-automation-step-executions \
```

```
    --automation-execution-id 73c8eef8-f4ee-4a05-820c-e354fEXAMPLE \
    --filters Key=StepExecutionId,Values=95e70479-cf20-4d80-8018-7e4e2EXAMPLE
```

Pour plus d'informations, consultez la section Exécution d'un flux de travail d'automatisation étape par étape (ligne de commande) dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous DescribeAutomationStepExecutionsà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple affiche des informations sur toutes les exécutions d'étapes actives et terminées dans un flux de travail d'automatisation.

```
Get-SSMAutomationStepExecution -AutomationExecutionId e1d2bad3-4567-8901-
ae23-456c7c8901be | Select-Object StepName, Action, StepStatus
```

Sortie :

```
StepName                  Action                    StepStatus
--------                  ------                    ----------
LaunchInstance            aws:runInstances          Success
OSCompatibilityCheck      aws:runCommand            Success
RunPreUpdateScript        aws:runCommand            Success
UpdateEC2Config           aws:runCommand            Cancelled
UpdateSSMAgent            aws:runCommand            Pending
UpdateAWSPVDriver         aws:runCommand            Pending
UpdateAWSEnaNetworkDriver aws:runCommand            Pending
UpdateAWSNVMe             aws:runCommand            Pending
InstallWindowsUpdates     aws:runCommand            Pending
RunPostUpdateScript       aws:runCommand            Pending
RunSysprepGeneralize      aws:runCommand            Pending
StopInstance              aws:changeInstanceState   Pending
CreateImage               aws:createImage           Pending
TerminateInstance         aws:changeInstanceState   Pending
```

- Pour plus de détails sur l'API, reportez-vous DescribeAutomationStepExecutionsà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeAvailablePatches`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribeAvailablePatches`.

CLI

AWS CLI

Pour obtenir les correctifs disponibles

L'`describe-available-patches`exemple suivant permet de récupérer des informations sur tous les correctifs disponibles pour Windows Server 2019 dont le niveau de gravité MSRC est Critique.

```
aws ssm describe-available-patches \
    --filters "Key=PRODUCT,Values=WindowsServer2019"
 "Key=MSRC_SEVERITY,Values=Critical"
```

Sortie :

```
{
    "Patches": [
        {
            "Id": "fe6bd8c2-3752-4c8b-ab3e-1a7ed08767ba",
            "ReleaseDate": 1544047205.0,
            "Title": "2018-11 Update for Windows Server 2019 for x64-based
 Systems (KB4470788)",
            "Description": "Install this update to resolve issues in Windows.
 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.",
            "ContentUrl": "https://support.microsoft.com/en-us/kb/4470788",
            "Vendor": "Microsoft",
            "ProductFamily": "Windows",
            "Product": "WindowsServer2019",
            "Classification": "SecurityUpdates",
            "MsrcSeverity": "Critical",
            "KbNumber": "KB4470788",
```

```
            "MsrcNumber": "",
            "Language": "All"
        },
        {
            "Id": "c96115e1-5587-4115-b851-22baa46a3f11",
            "ReleaseDate": 1549994410.0,
            "Title": "2019-02 Security Update for Adobe Flash Player for Windows
 Server 2019 for x64-based Systems (KB4487038)",
            "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.",
            "ContentUrl": "https://support.microsoft.com/en-us/kb/4487038",
            "Vendor": "Microsoft",
            "ProductFamily": "Windows",
            "Product": "WindowsServer2019",
            "Classification": "SecurityUpdates",
            "MsrcSeverity": "Critical",
            "KbNumber": "KB4487038",
            "MsrcNumber": "",
            "Language": "All"
        },
        ...
    ]
}
```

Pour obtenir des informations sur un correctif spécifique

L'`describe-available-patches`exemple suivant permet de récupérer des informations sur le correctif spécifié.

```
aws ssm describe-available-patches \
    --filters "Key=PATCH_ID,Values=KB4480979"
```

Sortie :

```
{
    "Patches": [
        {
            "Id": "680861e3-fb75-432e-818e-d72e5f2be719",
            "ReleaseDate": 1546970408.0,
```

```
            "Title": "2019-01 Security Update for Adobe Flash Player for Windows
 Server 2016 for x64-based Systems (KB4480979)",
            "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.",
            "ContentUrl": "https://support.microsoft.com/en-us/kb/4480979",
            "Vendor": "Microsoft",
            "ProductFamily": "Windows",
            "Product": "WindowsServer2016",
            "Classification": "SecurityUpdates",
            "MsrcSeverity": "Critical",
            "KbNumber": "KB4480979",
            "MsrcNumber": "",
            "Language": "All"
        }
    ]
}
```

Pour plus d'informations, reportez-vous à la section How Patch Manager Operations Work du Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, consultez la section DescribeAvailablePatches dans AWS CLI Command Reference.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple permet d'obtenir tous les correctifs disponibles pour Windows Server 2012 dont le niveau de gravité MSRC est Critique. La syntaxe utilisée dans cet exemple nécessite PowerShell la version 3 ou ultérieure.

```
$filter1 = @{Key="PRODUCT";Values=@("WindowsServer2012")}
$filter2 = @{Key="MSRC_SEVERITY";Values=@("Critical")}

Get-SSMAvailablePatch -Filter $filter1,$filter2
```

Sortie :

```
Classification : SecurityUpdates
```

```
ContentUrl     : https://support.microsoft.com/en-us/kb/2727528
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.
Id             : 1eb507be-2040-4eeb-803d-abc55700b715
KbNumber       : KB2727528
Language       : All
MsrcNumber     : MS12-072
MsrcSeverity   : Critical
Product        : WindowsServer2012
ProductFamily  : Windows
ReleaseDate    : 11/13/2012 6:00:00 PM
Title          : Security Update for Windows Server 2012 (KB2727528)
Vendor         : Microsoft
...
```

Exemple 2 : Avec PowerShell la version 2, vous devez utiliser New-Object pour créer chaque filtre.

```
$filter1 = New-Object
 Amazon.SimpleSystemsManagement.Model.PatchOrchestratorFilter
$filter1.Key = "PRODUCT"
$filter1.Values = "WindowsServer2012"
$filter2 = New-Object
 Amazon.SimpleSystemsManagement.Model.PatchOrchestratorFilter
$filter2.Key = "MSRC_SEVERITY"
$filter2.Values = "Critical"

Get-SSMAvailablePatch -Filter $filter1,$filter2
```

Exemple 3 : Cet exemple récupère toutes les mises à jour publiées au cours des 20 derniers jours et applicables aux produits correspondant WindowsServer à 2019

```
Get-SSMAvailablePatch | Where-Object ReleaseDate -ge (Get-Date).AddDays(-20)
 | Where-Object Product -eq "WindowsServer2019" | Select-Object ReleaseDate,
 Product, Title
```

Sortie :

```
ReleaseDate         Product           Title
```

```
-----------         -------           -----
4/9/2019 5:00:12 PM WindowsServer2019 2019-04 Security Update for Adobe Flash
 Player for Windows Server 2019 for x64-based Systems (KB4493478)
4/9/2019 5:00:06 PM WindowsServer2019 2019-04 Cumulative Update for Windows
 Server 2019 for x64-based Systems (KB4493509)
4/2/2019 5:00:06 PM WindowsServer2019 2019-03 Servicing Stack Update for Windows
 Server 2019 for x64-based Systems (KB4493510)
```

- Pour plus de détails sur l'API, consultez la section DescribeAvailablePatches in AWS Tools for PowerShell Cmdlet Reference.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeDocument`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribeDocument`.

CLI

AWS CLI

Pour afficher les détails d'un document

L'`describe-document`exemple suivant affiche les détails d'un document Systems Manager de votre AWS compte.

```
aws ssm describe-document \
    --name "Example"
```

Sortie :

```
{
    "Document": {
        "Hash":
 "fc2410281f40779e694a8b95975d0f9f316da8a153daa94e3d9921102EXAMPLE",
        "HashType": "Sha256",
        "Name": "Example",
        "Owner": "29884EXAMPLE",
        "CreatedDate": 1583257938.266,
```

```
        "Status": "Active",
        "DocumentVersion": "1",
        "Description": "Document Example",
        "Parameters": [
            {
                "Name": "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.",
                "DefaultValue": ""
            },
            {
                "Name": "InstanceId",
                "Type": "String",
                "Description": "(Required) The ID of the Amazon EC2 instance.",
                "DefaultValue": ""
            }
        ],
        "PlatformTypes": [
            "Windows",
            "Linux"
        ],
        "DocumentType": "Automation",
        "SchemaVersion": "0.3",
        "LatestVersion": "1",
        "DefaultVersion": "1",
        "DocumentFormat": "YAML",
        "Tags": []
    }
}
```

Pour plus d'informations, consultez la section Création de documents Systems Manager dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, voir DescribeDocumentla section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple renvoie des informations relatives à un document.

```
Get-SSMDocumentDescription -Name "RunShellScript"
```

Sortie :

```
CreatedDate     : 2/24/2017 5:25:13 AM
DefaultVersion  : 1
Description     : Run an updated script
DocumentType    : Command
DocumentVersion : 1
Hash            :
 f775e5df4904c6fa46686c4722fae9de1950dace25cd9608ff8d622046b68d9b
HashType        : Sha256
LatestVersion   : 1
Name            : RunShellScript
Owner           : 123456789012
Parameters      : {commands}
PlatformTypes   : {Linux}
SchemaVersion   : 2.0
Sha1            :
Status          : Active
```

- Pour plus de détails sur l'API, consultez la section DescribeDocumentRéférence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeDocumentPermission`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribeDocumentPermission`.

CLI

AWS CLI

Pour décrire les autorisations relatives aux documents

L'`describe-document-permission`exemple suivant affiche les détails des autorisations relatives à un document Systems Manager partagé publiquement.

```
aws ssm describe-document-permission \
    --name "Example" \
    --permission-type "Share"
```

Sortie :

```
{
    "AccountIds": [
        "all"
    ],
    "AccountSharingInfoList": [
        {
            "AccountId": "all",
            "SharedDocumentVersion": "$DEFAULT"
        }
    ]
}
```

Pour plus d'informations, consultez la section Partager un document Systems Manager dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, voir DescribeDocumentAutorisation dans AWS CLI la référence des commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple répertorie toutes les versions d'un document.

```
Get-SSMDocumentVersionList -Name "RunShellScript"
```

Sortie :

```
CreatedDate          DocumentVersion IsDefaultVersion Name
-----------          --------------- ---------------- ----
2/24/2017 5:25:13 AM 1               True             RunShellScript
```

- Pour plus de détails sur l'API, consultez la section DescribeDocumentAutorisation dans la référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeEffectiveInstanceAssociations`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribeEffectiveInstanceAssociations`.

CLI

AWS CLI

Pour obtenir des informations détaillées sur les associations efficaces pour une instance

L'`describe-effective-instance-associations`exemple suivant permet de récupérer des informations sur les associations efficaces pour une instance.

Commande :

```
aws ssm describe-effective-instance-associations --instance-id
 "i-1234567890abcdef0"
```

Sortie :

```
{
    "Associations": [
        {
            "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
            "InstanceId": "i-1234567890abcdef0",
            "Content": "{\n    \"schemaVersion\": \"1.2\",\n    \"description\":
 \"Update the Amazon SSM Agent to the latest version or specified version.\",\n
  \"parameters\": {\n        \"version\": {\n            \"default\": \"\",\n
       \"description\": \"(Optional) A specific version of the Amazon SSM Agent
 to install. If not specified, the agent will be updated to the latest version.
\",\n            \"type\": \"String\"\n        },\n        \"allowDowngrade
\": {\n            \"default\": \"false\",\n            \"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.\",\n            \"type
\": \"String\",\n            \"allowedValues\": [\n                \"true\",\n
```

```
                \"false\"\n            ]\n        }\n    },\n    \"runtimeConfig
\": {\n        \"aws:updateSsmAgent\": {\n            \"properties\": [\n
        {\n                \"agentName\": \"amazon-ssm-agent\",\n
  \"source\": \"https://s3.{Region}.amazonaws.com/amazon-ssm-{Region}/ssm-agent-
manifest.json\",\n                \"allowDowngrade\": \"{{ allowDowngrade }}\",\n
                \"targetVersion\": \"{{ version }}\"\n                }\n
    ]\n        }\n    }\n}\n",
            "AssociationVersion": "1"
        }
    ]
}
```

- Pour plus de détails sur l'API, reportez-vous DescribeEffectiveInstanceAssociationsà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple décrit les associations efficaces pour une instance.

```
Get-SSMEffectiveInstanceAssociationList -InstanceId "i-0000293ffd8c57862" -
MaxResult 5
```

Sortie :

```
AssociationId                        Content
-------------                        -------
d8617c07-2079-4c18-9847-1655fc2698b0 {...
```

Exemple 2 : Cet exemple affiche le contenu des associations efficaces pour une instance.

```
(Get-SSMEffectiveInstanceAssociationList -InstanceId "i-0000293ffd8c57862" -
MaxResult 5).Content
```

Sortie :

```
{
    "schemaVersion": "1.2",
    "description": "Update the Amazon SSM Agent to the latest version or
 specified version.",
```

```
    "parameters": {
        "version": {
            "default": "",
            "description": "(Optional) A specific version of the Amazon SSM Agent
 to install. If not specified, the agen
t will be updated to the latest version.",
            "type": "String"
        },
        "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"
            ]
        }
    },
    "runtimeConfig": {
        "aws:updateSsmAgent": {
            "properties": [
                {
                "agentName": "amazon-ssm-agent",
                "source": "https://s3.{Region}.amazonaws.com/amazon-ssm-{Region}/
ssm-agent-manifest.json",
                "allowDowngrade": "{{ allowDowngrade }}",
                "targetVersion": "{{ version }}"
                }
            ]
        }
    }
}
```

- Pour plus de détails sur l'API, reportez-vous DescribeEffectiveInstanceAssociationsà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **DescribeEffectivePatchesForPatchBaseline** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribeEffectivePatchesForPatchBaseline`.

CLI

AWS CLI

Exemple 1 : pour obtenir tous les correctifs définis par une ligne de base de correctifs personnalisée

L'`describe-effective-patches-for-patch-baseline`exemple suivant renvoie les correctifs définis par une ligne de base de correctifs personnalisée dans le AWS compte courant. Notez que pour une référence personnalisée, seul l'ID est requis pour`--baseline-id`.

```
aws ssm describe-effective-patches-for-patch-baseline \
    --baseline-id "pb-08b654cf9b9681f04"
```

Sortie :

```
{
    "EffectivePatches": [
        {
            "Patch": {
                "Id": "fe6bd8c2-3752-4c8b-ab3e-1a7ed08767ba",
                "ReleaseDate": 1544047205.0,
                "Title": "2018-11 Update for Windows Server 2019 for x64-based
 Systems (KB4470788)",
                "Description": "Install this update to resolve issues in Windows.
 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.",
                "ContentUrl": "https://support.microsoft.com/en-us/kb/4470788",
                "Vendor": "Microsoft",
                "ProductFamily": "Windows",
                "Product": "WindowsServer2019",
                "Classification": "SecurityUpdates",
                "MsrcSeverity": "Critical",
                "KbNumber": "KB4470788",
```

```
                "MsrcNumber": "",
                "Language": "All"
            },
            "PatchStatus": {
                "DeploymentStatus": "APPROVED",
                "ComplianceLevel": "CRITICAL",
                "ApprovalDate": 1544047205.0
            }
        },
        {
            "Patch": {
                "Id": "915a6b1a-f556-4d83-8f50-b2e75a9a7e58",
                "ReleaseDate": 1549994400.0,
                "Title": "2019-02 Cumulative Update for .NET Framework 3.5 and
 4.7.2 for Windows Server 2019 for x64 (KB4483452)",
                "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.",
                "ContentUrl": "https://support.microsoft.com/en-us/kb/4483452",
                "Vendor": "Microsoft",
                "ProductFamily": "Windows",
                "Product": "WindowsServer2019",
                "Classification": "SecurityUpdates",
                "MsrcSeverity": "Important",
                "KbNumber": "KB4483452",
                "MsrcNumber": "",
                "Language": "All"
            },
            "PatchStatus": {
                "DeploymentStatus": "APPROVED",
                "ComplianceLevel": "CRITICAL",
                "ApprovalDate": 1549994400.0
            }
        },
        ...
    ],
    "NextToken": "--token string truncated--"
}
```

Exemple 2 : pour obtenir tous les correctifs définis par une ligne de base de correctifs AWS gérée

L'`describe-effective-patches-for-patch-baseline`exemple suivant renvoie les correctifs définis par une ligne de base de correctifs AWS gérée. Notez que pour une ligne de base AWS gérée, l'ARN de référence complet est requis pour `--baseline-id`

```
aws ssm describe-effective-patches-for-patch-baseline \
    --baseline-id "arn:aws:ssm:us-east-2:733109147000:patchbaseline/
pb-020d361a05defe4ed"
```

Voir l'exemple 1 pour un exemple de sortie.

Pour plus d'informations, voir How Security Patches Are Selected dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous DescribeEffectivePatchesForPatchBaselineà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple répertorie toutes les lignes de base des correctifs, avec une liste de résultats maximale de 1.

```
Get-SSMEffectivePatchesForPatchBaseline -BaselineId "pb-0a2f1059b670ebd31" -
MaxResult 1
```

Sortie :

```
Patch                                       PatchStatus
-----                                       -----------
Amazon.SimpleSystemsManagement.Model.Patch
 Amazon.SimpleSystemsManagement.Model.PatchStatus
```

Exemple 2 : Cet exemple affiche l'état du correctif pour toutes les lignes de base des correctifs, avec une liste de résultats maximale de 1.

```
(Get-SSMEffectivePatchesForPatchBaseline -BaselineId "pb-0a2f1059b670ebd31" -
MaxResult 1).PatchStatus
```

Sortie :

```
ApprovalDate          DeploymentStatus
------------          ----------------
12/21/2010 6:00:00 PM APPROVED
```

- Pour plus de détails sur l'API, reportez-vous DescribeEffectivePatchesForPatchBaselineà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeInstanceAssociationsStatus`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribeInstanceAssociationsStatus`.

CLI

AWS CLI

Pour décrire le statut des associations d'une instance

Cet exemple montre les détails des associations associées à une instance.

Commande :

```
aws ssm describe-instance-associations-status --instance-id "i-1234567890abcdef0"
```

Sortie :

```
{
  "InstanceAssociationStatusInfos": [
```

```
      {
          "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
          "Name": "AWS-GatherSoftwareInventory",
          "DocumentVersion": "1",
          "AssociationVersion": "1",
          "InstanceId": "i-1234567890abcdef0",
          "ExecutionDate": 1550501886.0,
          "Status": "Success",
          "ExecutionSummary": "1 out of 1 plugin processed, 1 success, 0 failed,
 0 timedout, 0 skipped. ",
          "AssociationName": "Inventory-Association"
      },
      {
          "AssociationId": "5c5a31f6-6dae-46f9-944c-0123456789ab",
          "Name": "AWS-UpdateSSMAgent",
          "DocumentVersion": "1",
          "AssociationVersion": "1",
          "InstanceId": "i-1234567890abcdef0",
          "ExecutionDate": 1550505828.548,
          "Status": "Success",
          "DetailedStatus": "Success",
          "AssociationName": "UpdateSSMAgent"
      }
  ]
}
```

- Pour plus de détails sur l'API, voir DescribeInstanceAssociationsStatusla section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple montre les détails des associations associées à une instance.

```
Get-SSMInstanceAssociationsStatus -InstanceId "i-0000293ffd8c57862"
```

Sortie :

```
AssociationId    : d8617c07-2079-4c18-9847-1655fc2698b0
DetailedStatus   : Pending
DocumentVersion  : 1
```

```
ErrorCode        :
ExecutionDate    : 2/20/2015 8:31:11 AM
ExecutionSummary : temp_status_change
InstanceId       : i-0000293ffd8c57862
Name             : AWS-UpdateSSMAgent
OutputUrl        :
Status           : Pending
```

Exemple 2 : Cet exemple vérifie le statut de l'association d'instance pour l'identifiant d'instance donné et affiche ensuite le statut d'exécution de ces associations

```
Get-SSMInstanceAssociationsStatus -InstanceId i-012e3cb4df567e8aa | ForEach-
Object {Get-SSMAssociationExecution -AssociationId .AssociationId}
```

Sortie :

```
AssociationId         : 512a34a5-c678-1234-1234-12345678db9e
AssociationVersion    : 2
CreatedTime           : 3/2/2019 8:53:29 AM
DetailedStatus        :
ExecutionId           : 512a34a5-c678-1234-1234-12345678db9e
LastExecutionDate     : 1/1/0001 12:00:00 AM
ResourceCountByStatus : {Success=9}
Status                : Success
```

- Pour plus de détails sur l'API, consultez la section DescribeInstanceAssociationsStatusRéférence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeInstanceInformation`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribeInstanceInformation`.

CLI

AWS CLI

Exemple 1 : pour décrire les informations relatives à une instance gérée

L'`describe-instance-information`exemple suivant permet de récupérer les détails de chacune de vos instances gérées.

```
aws ssm describe-instance-information
```

Exemple 2 : pour décrire les informations relatives à une instance gérée spécifique

L'`describe-instance-information`exemple suivant montre les détails de l'instance gérée`i-028ea792daEXAMPLE`.

```
aws ssm describe-instance-information \
    --filters "Key=InstanceIds,Values=i-028ea792daEXAMPLE"
```

Exemple 3 : pour décrire les informations relatives aux instances gérées avec une clé de balise spécifique

L'`describe-instance-information`exemple suivant montre les détails des instances gérées dotées de la clé de balise`DEV`.

```
aws ssm describe-instance-information \
    --filters "Key=tag-key,Values=DEV"
```

Sortie :

```
{
    "InstanceInformationList": [
        {
            "InstanceId": "i-028ea792daEXAMPLE",
            "PingStatus": "Online",
            "LastPingDateTime": 1582221233.421,
            "AgentVersion": "2.3.842.0",
            "IsLatestVersion": true,
            "PlatformType": "Linux",
            "PlatformName": "SLES",
            "PlatformVersion": "15.1",
```

```
            "ResourceType": "EC2Instance",
            "IPAddress": "192.0.2.0",
            "ComputerName": "ip-198.51.100.0.us-east-2.compute.internal",
            "AssociationStatus": "Success",
            "LastAssociationExecutionDate": 1582220806.0,
            "LastSuccessfulAssociationExecutionDate": 1582220806.0,
            "AssociationOverview": {
                "DetailedStatus": "Success",
                "InstanceAssociationStatusAggregatedCount": {
                    "Success": 2
                }
            }
        }
    ]
}
```

Pour plus d'informations, consultez la section Instances gérées dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, consultez les DescribeInstanceinformations dans le manuel de référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple montre les détails de chacune de vos instances.

```
Get-SSMInstanceInformation
```

Sortie :

```
ActivationId                        :
AgentVersion                        : 2.0.672.0
AssociationOverview                 :
 Amazon.SimpleSystemsManagement.Model.InstanceAggregatedAssociationOverview
AssociationStatus                   : Success
ComputerName                        : ip-172-31-44-222.us-
west-2.compute.internal
IamRole                             :
InstanceId                          : i-0cb2b964d3e14fd9f
IPAddress                           : 172.31.44.222
```

```
IsLatestVersion                         : True
LastAssociationExecutionDate            : 2/24/2017 3:18:09 AM
LastPingDateTime                        : 2/24/2017 3:35:03 AM
LastSuccessfulAssociationExecutionDate : 2/24/2017 3:18:09 AM
Name                                    :
PingStatus                              : ConnectionLost
PlatformName                            : Amazon Linux AMI
PlatformType                            : Linux
PlatformVersion                         : 2016.09
RegistrationDate                        : 1/1/0001 12:00:00 AM
ResourceType                            : EC2Instance
```

Exemple 2 : Cet exemple montre comment utiliser le paramètre -Filter pour filtrer les résultats uniquement en fonction des instances de AWS Systems Manager situées dans une région **`us-east-1`** avec un **`AgentVersion`** de**`2.2.800.0`**. Vous trouverez une liste des valeurs de clé -Filter valides dans la rubrique de référence de l' InstanceInformation API (https://docs.aws.amazon.com/systems-manager/latest/APIReference/API_ InstanceInformation .html #systemsmanager -Type- -InstanceInformation). ActivationId

```
$Filters = @{
    Key="AgentVersion"
    Values="2.2.800.0"
}
Get-SSMInstanceInformation -Region us-east-1 -Filter $Filters
```

Sortie :

```
ActivationId                            :
AgentVersion                            : 2.2.800.0
AssociationOverview                     :
 Amazon.SimpleSystemsManagement.Model.InstanceAggregatedAssociationOverview
AssociationStatus                       : Success
ComputerName                            : EXAMPLE-EXAMPLE.WORKGROUP
IamRole                                 :
InstanceId                              : i-EXAMPLEb0792d98ce
IPAddress                               : 10.0.0.01
IsLatestVersion                         : False
LastAssociationExecutionDate            : 8/16/2018 12:02:50 AM
LastPingDateTime                        : 8/16/2018 7:40:27 PM
LastSuccessfulAssociationExecutionDate : 8/16/2018 12:02:50 AM
Name                                    :
PingStatus                              : Online
```

```
PlatformName                          : Microsoft Windows Server 2016 Datacenter
PlatformType                          : Windows
PlatformVersion                       : 10.0.14393
RegistrationDate                      : 1/1/0001 12:00:00 AM
ResourceType                          : EC2Instance

ActivationId                          :
AgentVersion                          : 2.2.800.0
AssociationOverview                   :
 Amazon.SimpleSystemsManagement.Model.InstanceAggregatedAssociationOverview
AssociationStatus                     : Success
ComputerName                          : EXAMPLE-EXAMPLE.WORKGROUP
IamRole                               :
InstanceId                            : i-EXAMPLEac7501d023
IPAddress                             : 10.0.0.02
IsLatestVersion                       : False
LastAssociationExecutionDate          : 8/16/2018 12:00:20 AM
LastPingDateTime                      : 8/16/2018 7:40:35 PM
LastSuccessfulAssociationExecutionDate : 8/16/2018 12:00:20 AM
Name                                  :
PingStatus                            : Online
PlatformName                          : Microsoft Windows Server 2016 Datacenter
PlatformType                          : Windows
PlatformVersion                       : 10.0.14393
RegistrationDate                      : 1/1/0001 12:00:00 AM
ResourceType                          : EC2Instance
```

Exemple 3 : Cet exemple montre comment utiliser le InstanceInformationFilterList paramètre - pour filtrer les résultats uniquement en fonction des instances **`PlatformTypes`** de AWS Systems Manager situées dans **`us-east-1`** une région avec **`Windows`** ou**`Linux`**. Vous trouverez une liste des valeurs InstanceInformationFilterList clés valides dans la rubrique de référence de l' InstanceInformationFilter API (https://docs.aws.amazon.com/systems-manager/latest/APIReference/API_ InstanceInformation Filter.html).

```
$Filters = @{
   Key="PlatformTypes"
   ValueSet=("Windows","Linux")
}
Get-SSMInstanceInformation -Region us-east-1 -InstanceInformationFilterList
 $Filters
```

Sortie :

```
ActivationId                          :
AgentVersion                          : 2.2.800.0
AssociationOverview                   :
 Amazon.SimpleSystemsManagement.Model.InstanceAggregatedAssociationOverview
AssociationStatus                     : Success
ComputerName                          : EXAMPLE-EXAMPLE.WORKGROUP
IamRole                               :
InstanceId                            : i-EXAMPLEb0792d98ce
IPAddress                             : 10.0.0.27
IsLatestVersion                       : False
LastAssociationExecutionDate          : 8/16/2018 12:02:50 AM
LastPingDateTime                      : 8/16/2018 7:40:27 PM
LastSuccessfulAssociationExecutionDate : 8/16/2018 12:02:50 AM
Name                                  :
PingStatus                            : Online
PlatformName                          : Ubuntu Server 18.04 LTS
PlatformType                          : Linux
PlatformVersion                       : 18.04
RegistrationDate                      : 1/1/0001 12:00:00 AM
ResourceType                          : EC2Instance

ActivationId                          :
AgentVersion                          : 2.2.800.0
AssociationOverview                   :
 Amazon.SimpleSystemsManagement.Model.InstanceAggregatedAssociationOverview
AssociationStatus                     : Success
ComputerName                          : EXAMPLE-EXAMPLE.WORKGROUP
IamRole                               :
InstanceId                            : i-EXAMPLEac7501d023
IPAddress                             : 10.0.0.100
IsLatestVersion                       : False
LastAssociationExecutionDate          : 8/16/2018 12:00:20 AM
LastPingDateTime                      : 8/16/2018 7:40:35 PM
LastSuccessfulAssociationExecutionDate : 8/16/2018 12:00:20 AM
Name                                  :
PingStatus                            : Online
PlatformName                          : Microsoft Windows Server 2016 Datacenter
PlatformType                          : Windows
PlatformVersion                       : 10.0.14393
RegistrationDate                      : 1/1/0001 12:00:00 AM
ResourceType                          : EC2Instance
```

Exemple 4 : Cet exemple répertorie les instances gérées par SSM et les exportations InstanceId PingStatus, LastPingDateTime et PlatformName vers un fichier csv.

```
Get-SSMInstanceInformation | Select-Object InstanceId, PingStatus,
 LastPingDateTime, PlatformName | Export-Csv Instance-details.csv -
NoTypeInformation
```

- Pour plus de détails sur l'API, consultez les DescribeInstanceinformations contenues dans le manuel de référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeInstancePatchStates`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribeInstancePatchStates`.

CLI

AWS CLI

Pour obtenir les états récapitulatifs des correctifs pour les instances

Cet `describe-instance-patch-states` exemple permet d'obtenir les états récapitulatifs des correctifs pour une instance.

```
aws ssm describe-instance-patch-states \
    --instance-ids "i-1234567890abcdef0"
```

Sortie :

```
{
    "InstancePatchStates": [
        {
            "InstanceId": "i-1234567890abcdef0",
            "PatchGroup": "my-patch-group",
            "BaselineId": "pb-0713accee01234567",
```

```
            "SnapshotId": "521c3536-930c-4aa9-950e-01234567abcd",
            "CriticalNonCompliantCount": 2,
            "SecurityNonCompliantCount": 2,
            "OtherNonCompliantCount": 1,
            "InstalledCount": 123,
            "InstalledOtherCount": 334,
            "InstalledPendingRebootCount": 0,
            "InstalledRejectedCount": 0,
            "MissingCount": 1,
            "FailedCount": 2,
            "UnreportedNotApplicableCount": 11,
            "NotApplicableCount": 2063,
            "OperationStartTime": "2021-05-03T11:00:56-07:00",
            "OperationEndTime": "2021-05-03T11:01:09-07:00",
            "Operation": "Scan",
            "LastNoRebootInstallOperationTime": "2020-06-14T12:17:41-07:00",
            "RebootOption": "RebootIfNeeded"
        }
    ]
}
```

Pour plus d'informations, reportez-vous à la section About Patch Compliance dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous DescribeInstancePatchStatesà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple obtient les états récapitulatifs des correctifs pour une instance.

```
Get-SSMInstancePatchState -InstanceId "i-08ee91c0b17045407"
```

Exemple 2 : Cet exemple obtient les états récapitulatifs des correctifs pour deux instances.

```
Get-SSMInstancePatchState -InstanceId "i-08ee91c0b17045407","i-09a618aec652973a9"
```

- Pour plus de détails sur l'API, reportez-vous DescribeInstancePatchStatesà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **DescribeInstancePatchStatesForPatchGroup** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribeInstancePatchStatesForPatchGroup`.

CLI

AWS CLI

Exemple 1 : pour obtenir les états d'instance d'un groupe de correctifs

L'`describe-instance-patch-states-for-patch-group`exemple suivant permet de récupérer des informations sur les états récapitulatifs des correctifs par instance pour le groupe de correctifs spécifié.

```
aws ssm describe-instance-patch-states-for-patch-group \
    --patch-group "Production"
```

Sortie :

```
{
    "InstancePatchStates": [
        {
            "InstanceId": "i-02573cafcfEXAMPLE",
            "PatchGroup": "Production",
            "BaselineId": "pb-0c10e65780EXAMPLE",
            "SnapshotId": "a3f5ff34-9bc4-4d2c-a665-4d1c1EXAMPLE",
            "OwnerInformation": "",
            "InstalledCount": 32,
            "InstalledOtherCount": 1,
            "InstalledPendingRebootCount": 0,
            "InstalledRejectedCount": 0,
            "MissingCount": 2,
            "FailedCount": 0,
            "UnreportedNotApplicableCount": 2671,
            "NotApplicableCount": 400,
            "OperationStartTime": "2021-08-04T11:03:50.590000-07:00",
```

```
            "OperationEndTime": "2021-08-04T11:04:21.555000-07:00",
            "Operation": "Scan",
            "RebootOption": "NoReboot",
            "CriticalNonCompliantCount": 0,
            "SecurityNonCompliantCount": 1,
            "OtherNonCompliantCount": 0
        },
        {
            "InstanceId": "i-0471e04240EXAMPLE",
            "PatchGroup": "Production",
            "BaselineId": "pb-09ca3fb51fEXAMPLE",
            "SnapshotId": "05d8ffb0-1bbe-4812-ba2d-d9b7bEXAMPLE",
            "OwnerInformation": "",
            "InstalledCount": 32,
            "InstalledOtherCount": 1,
            "InstalledPendingRebootCount": 0,
            "InstalledRejectedCount": 0,
            "MissingCount": 2,
            "FailedCount": 0,
            "UnreportedNotApplicableCount": 2671,
            "NotApplicableCount": 400,
            "OperationStartTime": "2021-08-04T22:06:20.340000-07:00",
            "OperationEndTime": "2021-08-04T22:07:11.220000-07:00",
            "Operation": "Scan",
            "RebootOption": "NoReboot",
            "CriticalNonCompliantCount": 0,
            "SecurityNonCompliantCount": 1,
            "OtherNonCompliantCount": 0
        }
    ]
}
```

Exemple 2 : pour obtenir les états d'instance d'un groupe de correctifs contenant plus de cinq correctifs manquants

L'`describe-instance-patch-states-for-patch-group`exemple suivant permet de récupérer des informations sur les états récapitulatifs des correctifs pour le groupe de correctifs spécifié pour les instances comportant plus de cinq correctifs manquants.

```
aws ssm describe-instance-patch-states-for-patch-group \
    --filters Key=MissingCount,Type=GreaterThan,Values=5 \
    --patch-group "Production"
```

Sortie :

```
{
    "InstancePatchStates": [
        {
            "InstanceId": "i-02573cafcfEXAMPLE",
            "PatchGroup": "Production",
            "BaselineId": "pb-0c10e65780EXAMPLE",
            "SnapshotId": "a3f5ff34-9bc4-4d2c-a665-4d1c1EXAMPLE",
            "OwnerInformation": "",
            "InstalledCount": 46,
            "InstalledOtherCount": 4,
            "InstalledPendingRebootCount": 1,
            "InstalledRejectedCount": 1,
            "MissingCount": 7,
            "FailedCount": 0,
            "UnreportedNotApplicableCount": 232,
            "NotApplicableCount": 654,
            "OperationStartTime": "2021-08-04T11:03:50.590000-07:00",
            "OperationEndTime": "2021-08-04T11:04:21.555000-07:00",
            "Operation": "Scan",
            "RebootOption": "NoReboot",
            "CriticalNonCompliantCount": 0,
            "SecurityNonCompliantCount": 1,
            "OtherNonCompliantCount": 1
        }
    ]
}
```

Exemple 3 : pour obtenir les états d'instance d'un groupe de correctifs comportant moins de dix instances nécessitant un redémarrage

L'`describe-instance-patch-states-for-patch-group`exemple suivant permet de récupérer des informations sur les états récapitulatifs des correctifs pour le groupe de correctifs spécifié pour les instances dont moins de dix instances nécessitent un redémarrage.

```
aws ssm describe-instance-patch-states-for-patch-group \
    --filters Key=InstalledPendingRebootCount,Type=LessThan,Values=10 \
    --patch-group "Production"
```

Sortie :

```
{
    "InstancePatchStates": [
        {
            "InstanceId": "i-02573cafcfEXAMPLE",
            "BaselineId": "pb-0c10e65780EXAMPLE",
            "SnapshotId": "a3f5ff34-9bc4-4d2c-a665-4d1c1EXAMPLE",
            "PatchGroup": "Production",
            "OwnerInformation": "",
            "InstalledCount": 32,
            "InstalledOtherCount": 1,
            "InstalledPendingRebootCount": 4,
            "InstalledRejectedCount": 0,
            "MissingCount": 2,
            "FailedCount": 0,
            "UnreportedNotApplicableCount": 846,
            "NotApplicableCount": 212,
            "OperationStartTime": "2021-08-046T11:03:50.590000-07:00",
            "OperationEndTime": "2021-08-06T11:04:21.555000-07:00",
            "Operation": "Scan",
            "RebootOption": "NoReboot",
            "CriticalNonCompliantCount": 0,
            "SecurityNonCompliantCount": 1,
            "OtherNonCompliantCount": 0
        }
    ]
}
```

Pour plus d'informations, consultez la section Comprendre les valeurs d'état de conformité des correctifs dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, consultez la section DescribeInstancePatchStatesForPatchGroup in AWS CLI Command Reference.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple obtient les états récapitulatifs des correctifs par instance pour un groupe de correctifs.

```
Get-SSMInstancePatchStatesForPatchGroup -PatchGroup "Production"
```

- Pour plus de détails sur l'API, consultez la section DescribeInstancePatchStatesForPatchGroup in AWS Tools for PowerShell Cmdlet Reference.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeInstancePatches`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribeInstancePatches`.

CLI

AWS CLI

Exemple 1 : pour obtenir les détails de l'état du correctif pour une instance

L'`describe-instance-patches`exemple suivant permet de récupérer des informations sur les correctifs pour l'instance spécifiée.

```
aws ssm describe-instance-patches \
    --instance-id "i-1234567890abcdef0"
```

Sortie :

```
{
    "Patches": [
        {
            "Title": "2019-01 Security Update for Adobe Flash Player for Windows
 Server 2016 for x64-based Systems (KB4480979)",
            "KBId": "KB4480979",
            "Classification": "SecurityUpdates",
            "Severity": "Critical",
            "State": "Installed",
            "InstalledTime": "2019-01-09T00:00:00+00:00"
        },
        {
            "Title": "",
            "KBId": "KB4481031",
            "Classification": "",
```

```
            "Severity": "",
            "State": "InstalledOther",
            "InstalledTime": "2019-02-08T00:00:00+00:00"
        },
        ...
    ],
    "NextToken": "--token string truncated--"
}
```

Exemple 2 : pour obtenir la liste des correctifs à l'état manquant pour une instance

L'`describe-instance-patches`exemple suivant récupère des informations sur les correctifs dont l'état est manquant pour l'instance spécifiée.

```
aws ssm describe-instance-patches \
    --instance-id "i-1234567890abcdef0" \
    --filters Key=State,Values=Missing
```

Sortie :

```
{
    "Patches": [
        {
            "Title": "Windows Malicious Software Removal Tool x64 - February 2019
 (KB890830)",
            "KBId": "KB890830",
            "Classification": "UpdateRollups",
            "Severity": "Unspecified",
            "State": "Missing",
            "InstalledTime": "1970-01-01T00:00:00+00:00"
        },
        ...
    ],
    "NextToken": "--token string truncated--"
}
```

Pour plus d'informations, consultez la section À propos des états de conformité des correctifs dans le guide de l'utilisateur de AWS Systems Manager.

Exemple 3 : pour obtenir la liste des correctifs installés depuis une spécification InstalledTime pour une instance

L'`describe-instance-patches`exemple suivant extrait des informations sur les correctifs installés depuis une date spécifiée pour l'instance spécifiée en combinant l'utilisation de `--filters` et`--query`.

```
aws ssm describe-instance-patches \
    --instance-id "i-1234567890abcdef0" \
    --filters Key=State,Values=Installed \
    --query "Patches[?InstalledTime >= `2023-01-01T16:00:00`]"
```

Sortie :

```
{
    "Patches": [
        {
            "Title": "2023-03 Cumulative Update for Windows Server 2019 (1809)
 for x64-based Systems (KB5023702)",
            "KBId": "KB5023702",
            "Classification": "SecurityUpdates",
            "Severity": "Critical",
            "State": "Installed",
            "InstalledTime": "2023-03-16T11:00:00+00:00"
        },
        ...
    ],
    "NextToken": "--token string truncated--"
}
```

- Pour plus de détails sur l'API, consultez la section DescribeInstancePatches dans AWS CLI Command Reference.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple permet d'obtenir les détails de conformité des correctifs pour une instance.

```
Get-SSMInstancePatch -InstanceId "i-08ee91c0b17045407"
```

- Pour plus de détails sur l'API, consultez la section DescribeInstancePatches in AWS Tools for PowerShell Cmdlet Reference.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeMaintenanceWindowExecutionTaskInvocations`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribeMaintenanceWindowExecutionTaskInvocations`.

CLI

AWS CLI

Pour effectuer les invocations de tâches spécifiques pour une fenêtre de maintenance, exécution de tâches

L'`describe-maintenance-window-execution-task-invocations`exemple suivant répertorie les appels pour la tâche spécifiée exécutés dans le cadre de l'exécution de la fenêtre de maintenance spécifiée.

```
aws ssm describe-maintenance-window-execution-task-invocations \
    --window-execution-id "518d5565-5969-4cca-8f0e-da3b2a638355" \
    --task-id "ac0c6ae1-daa3-4a89-832e-d384503b6586"
```

Sortie :

```
{
    "WindowExecutionTaskInvocationIdentities": [
        {
            "Status": "SUCCESS",
            "Parameters": "{\"documentName\":\"AWS-RunShellScript\",
\"instanceIds\":[\"i-0000293ffd8c57862\"],\"parameters\":{\"commands\":[\"df\"]},
\"maxConcurrency\":\"1\",\"maxErrors\":\"1\"}",
            "InvocationId": "e274b6e1-fe56-4e32-bd2a-8073c6381d8b",
            "StartTime": 1487692834.723,
            "EndTime": 1487692834.871,
            "WindowExecutionId": "518d5565-5969-4cca-8f0e-da3b2a638355",
            "TaskExecutionId": "ac0c6ae1-daa3-4a89-832e-d384503b6586"
```

```
        }
    ]
}
```

Pour plus d'informations, consultez la section Afficher les informations sur les tâches et les exécutions de tâches (AWS CLI) dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous DescribeMaintenanceWindowExecutionTaskInvocationsà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple répertorie les appels pour une tâche exécutée dans le cadre de l'exécution d'une fenêtre de maintenance.

```
Get-SSMMaintenanceWindowExecutionTaskInvocationList -TaskId "ac0c6ae1-
daa3-4a89-832e-d384503b6586" -WindowExecutionId "518d5565-5969-4cca-8f0e-
da3b2a638355"
```

Sortie :

```
EndTime           : 2/21/2017 4:00:34 PM
ExecutionId       :
InvocationId      : e274b6e1-fe56-4e32-bd2a-8073c6381d8b
OwnerInformation  :
Parameters        : {"documentName":"AWS-RunShellScript","instanceIds":
["i-0000293ffd8c57862"],"parameters":{"commands":["df"]},"maxConcurrency":"1",
                    "maxErrors":"1"}
StartTime         : 2/21/2017 4:00:34 PM
Status            : FAILED
StatusDetails     : The instance IDs list contains an invalid entry.
TaskExecutionId   : ac0c6ae1-daa3-4a89-832e-d384503b6586
WindowExecutionId : 518d5565-5969-4cca-8f0e-da3b2a638355
WindowTargetId    :
```

- Pour plus de détails sur l'API, consultez la section DescribeMaintenanceWindowExecutionTaskInvocationsRéférence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeMaintenanceWindowExecutionTasks`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribeMaintenanceWindowExecutionTasks`.

CLI

AWS CLI

Pour répertorier toutes les tâches associées à l'exécution d'une fenêtre de maintenance

L'`ssm describe-maintenance-window-execution-tasks`exemple suivant répertorie les tâches associées à l'exécution de la fenêtre de maintenance spécifiée.

```
aws ssm describe-maintenance-window-execution-tasks \
    --window-execution-id "518d5565-5969-4cca-8f0e-da3b2EXAMPLE"
```

Sortie :

```
{
    "WindowExecutionTaskIdentities": [
        {
            "Status": "SUCCESS",
            "TaskArn": "AWS-RunShellScript",
            "StartTime": 1487692834.684,
            "TaskType": "RUN_COMMAND",
            "EndTime": 1487692835.005,
            "WindowExecutionId": "518d5565-5969-4cca-8f0e-da3b2EXAMPLE",
            "TaskExecutionId": "ac0c6ae1-daa3-4a89-832e-d3845EXAMPLE"
        }
    ]
}
```

Pour plus d'informations, consultez la section Afficher les informations sur les tâches et les exécutions de tâches (AWS CLI) dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, consultez la section DescribeMaintenanceWindowExecutionTâches dans AWS CLI la référence des commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple répertorie les tâches associées à l'exécution d'une fenêtre de maintenance.

```
Get-SSMMaintenanceWindowExecutionTaskList -WindowExecutionId
 "518d5565-5969-4cca-8f0e-da3b2a638355"
```

Sortie :

```
EndTime           : 2/21/2017 4:00:35 PM
StartTime         : 2/21/2017 4:00:34 PM
Status            : SUCCESS
TaskArn           : AWS-RunShellScript
TaskExecutionId   : ac0c6ae1-daa3-4a89-832e-d384503b6586
TaskType          : RUN_COMMAND
WindowExecutionId : 518d5565-5969-4cca-8f0e-da3b2a638355
```

- Pour plus de détails sur l'API, consultez la section DescribeMaintenanceWindowExecutionTâches dans la référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeMaintenanceWindowExecutions`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribeMaintenanceWindowExecutions`.

CLI

AWS CLI

Exemple 1 : pour répertorier toutes les exécutions pour une fenêtre de maintenance

L'`describe-maintenance-window-executions`exemple suivant répertorie toutes les exécutions pour la fenêtre de maintenance spécifiée.

```
aws ssm describe-maintenance-window-executions \
    --window-id "mw-ab12cd34eEXAMPLE"
```

Sortie :

```
{
    "WindowExecutions": [
        {
            "WindowId": "mw-ab12cd34eEXAMPLE",
            "WindowExecutionId": "6027b513-64fe-4cf0-be7d-1191aEXAMPLE",
            "Status": "IN_PROGRESS",
            "StartTime": "2021-08-04T11:00:00.000000-07:00"

        },
        {
            "WindowId": "mw-ab12cd34eEXAMPLE",
            "WindowExecutionId": "ff75b750-4834-4377-8f61-b3cadEXAMPLE",
            "Status": "SUCCESS",
            "StartTime": "2021-08-03T11:00:00.000000-07:00",
            "EndTime": "2021-08-03T11:37:21.450000-07:00"
        },
        {
            "WindowId": "mw-ab12cd34eEXAMPLE",
            "WindowExecutionId": "9fac7dd9-ff21-42a5-96ad-bbc4bEXAMPLE",
            "Status": "FAILED",
            "StatusDetails": "One or more tasks in the orchestration failed.",
            "StartTime": "2021-08-02T11:00:00.000000-07:00",
            "EndTime": "2021-08-02T11:22:36.190000-07:00"
        }
    ]
}
```

Exemple 2 : Pour répertorier toutes les exécutions pour une fenêtre de maintenance avant une date spécifiée

L'`describe-maintenance-window-executions`exemple suivant répertorie toutes les exécutions pour la fenêtre de maintenance spécifiée avant la date spécifiée.

```
aws ssm describe-maintenance-window-executions \
    --window-id "mw-ab12cd34eEXAMPLE" \
    --filters "Key=ExecutedBefore,Values=2021-08-03T00:00:00Z"
```

Sortie :

```
{
    "WindowExecutions": [
        {
        "WindowId": "mw-ab12cd34eEXAMPLE",
        "WindowExecutionId": "9fac7dd9-ff21-42a5-96ad-bbc4bEXAMPLE",
        "Status": "FAILED",
        "StatusDetails": "One or more tasks in the orchestration failed.",
        "StartTime": "2021-08-02T11:00:00.000000-07:00",
        "EndTime": "2021-08-02T11:22:36.190000-07:00"
    }
    ]
}
```

Exemple 3 : pour répertorier toutes les exécutions pour une fenêtre de maintenance après une date spécifiée

L'`describe-maintenance-window-executions`exemple suivant répertorie toutes les exécutions pour la fenêtre de maintenance spécifiée après la date spécifiée.

```
aws ssm describe-maintenance-window-executions \
    --window-id "mw-ab12cd34eEXAMPLE" \
    --filters "Key=ExecutedAfter,Values=2021-08-04T00:00:00Z"
```

Sortie :

```
{
    "WindowExecutions": [
        {
        "WindowId": "mw-ab12cd34eEXAMPLE",
        "WindowExecutionId": "6027b513-64fe-4cf0-be7d-1191aEXAMPLE",
        "Status": "IN_PROGRESS",
        "StartTime": "2021-08-04T11:00:00.000000-07:00"
```

```
        }
    ]
}
```

Pour plus d'informations, consultez la section Afficher les informations relatives aux tâches et aux exécutions de tâches (AWS CLI) dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous DescribeMaintenanceWindowExecutionsà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple répertorie toutes les exécutions pour une fenêtre de maintenance.

```
Get-SSMMaintenanceWindowExecutionList -WindowId "mw-03eb9db42890fb82d"
```

Sortie :

```
EndTime           : 2/20/2017 6:30:17 PM
StartTime         : 2/20/2017 6:30:16 PM
Status            : FAILED
StatusDetails     : One or more tasks in the orchestration failed.
WindowExecutionId : 6f3215cf-4101-4fa0-9b7b-9523269599c7
WindowId          : mw-03eb9db42890fb82d
```

Exemple 2 : Cet exemple répertorie toutes les exécutions pour une fenêtre de maintenance avant une date spécifiée.

```
$option1 = @{Key="ExecutedBefore";Values=@("2016-11-04T05:00:00Z")}
Get-SSMMaintenanceWindowExecutionList -WindowId "mw-03eb9db42890fb82d" -Filter
 $option1
```

Exemple 3 : Cet exemple répertorie toutes les exécutions pour une fenêtre de maintenance après une date spécifiée.

```
$option1 = @{Key="ExecutedAfter";Values=@("2016-11-04T05:00:00Z")}
Get-SSMMaintenanceWindowExecutionList -WindowId "mw-03eb9db42890fb82d" -Filter
 $option1
```

- Pour plus de détails sur l'API, reportez-vous DescribeMaintenanceWindowExecutionsà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeMaintenanceWindowTargets`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribeMaintenanceWindowTargets`.

CLI

AWS CLI

Exemple 1 : pour répertorier toutes les cibles d'une fenêtre de maintenance

L'`describe-maintenance-window-targets`exemple suivant répertorie toutes les cibles d'une fenêtre de maintenance.

```
aws ssm describe-maintenance-window-targets \
    --window-id "mw-06cf17cbefEXAMPLE"
```

Sortie :

```
{
    "Targets": [
        {
            "ResourceType": "INSTANCE",
            "OwnerInformation": "Single instance",
            "WindowId": "mw-06cf17cbefEXAMPLE",
            "Targets": [
                {
                    "Values": [
                        "i-0000293ffdEXAMPLE"
                    ],
                    "Key": "InstanceIds"
                }
```

```
            ],
            "WindowTargetId": "350d44e6-28cc-44e2-951f-4b2c9EXAMPLE"
        },
        {
            "ResourceType": "INSTANCE",
            "OwnerInformation": "Two instances in a list",
            "WindowId": "mw-06cf17cbefEXAMPLE",
            "Targets": [
                {
                    "Values": [
                        "i-0000293ffdEXAMPLE",
                        "i-0cb2b964d3EXAMPLE"
                    ],
                    "Key": "InstanceIds"
                }
            ],
            "WindowTargetId": "e078a987-2866-47be-bedd-d9cf4EXAMPLE"
        }
    ]
}
```

Exemple 2 : pour répertorier toutes les cibles d'une fenêtre de maintenance correspondant à une valeur d'information spécifique sur le propriétaire

Cet `describe-maintenance-window-targets` exemple répertorie toutes les cibles d'une fenêtre de maintenance avec une valeur spécifique.

```
aws ssm describe-maintenance-window-targets \
    --window-id "mw-0ecb1226ddEXAMPLE" \
    --filters "Key=OwnerInformation,Values=CostCenter1"
```

Sortie :

```
{
    "Targets": [
        {
            "WindowId": "mw-0ecb1226ddEXAMPLE",
            "WindowTargetId": "da89dcc3-7f9c-481d-ba2b-edcb7d0057f9",
            "ResourceType": "INSTANCE",
            "Targets": [
                {
                    "Key": "tag:Environment",
```

```
                    "Values": [
                        "Prod"
                    ]
                }
            ],
            "OwnerInformation": "CostCenter1",
            "Name": "ProdTarget1"
        }
    ]
}
```

Pour plus d'informations, consultez la section Afficher les informations relatives à la maintenance de Windows (AWS CLI) dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous DescribeMaintenanceWindowTargetsà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple répertorie toutes les cibles d'une fenêtre de maintenance.

```
Get-SSMMaintenanceWindowTarget -WindowId "mw-06cf17cbefcb4bf4f"
```

Sortie :

```
OwnerInformation : Single instance
ResourceType     : INSTANCE
Targets          : {InstanceIds}
WindowId         : mw-06cf17cbefcb4bf4f
WindowTargetId   : 350d44e6-28cc-44e2-951f-4b2c985838f6

OwnerInformation : Two instances in a list
ResourceType     : INSTANCE
Targets          : {InstanceIds}
WindowId         : mw-06cf17cbefcb4bf4f
WindowTargetId   : e078a987-2866-47be-bedd-d9cf49177d3a
```

- Pour plus de détails sur l'API, reportez-vous DescribeMaintenanceWindowTargetsà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeMaintenanceWindowTasks`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribeMaintenanceWindowTasks`.

CLI

AWS CLI

Exemple 1 : pour répertorier toutes les tâches d'une fenêtre de maintenance

L'`describe-maintenance-window-tasks`exemple suivant répertorie toutes les tâches correspondant à la fenêtre de maintenance spécifiée.

```
aws ssm describe-maintenance-window-tasks \
    --window-id "mw-06cf17cbefEXAMPLE"
```

Sortie :

```
{
    "Tasks": [
        {
            "WindowId": "mw-06cf17cbefEXAMPLE",
            "WindowTaskId": "018b31c3-2d77-4b9e-bd48-c91edEXAMPLE",
            "TaskArn": "AWS-RestartEC2Instance",
            "TaskParameters": {},
            "Type": "AUTOMATION",
            "Description": "Restarting EC2 Instance for maintenance",
            "MaxConcurrency": "1",
            "MaxErrors": "1",
            "Name": "My-Automation-Example-Task",
            "Priority": 0,
            "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
            "Targets": [
                {
                    "Key": "WindowTargetIds",
                    "Values": [
```

```
                        "da89dcc3-7f9c-481d-ba2b-edcb7EXAMPLE"
                    ]
                }
            ]
        },
        {
            "WindowId": "mw-06cf17cbefEXAMPLE",
            "WindowTaskId": "1943dee0-0a17-4978-9bf4-3cc2fEXAMPLE",
            "TaskArn": "AWS-DisableS3BucketPublicReadWrite",
            "TaskParameters": {},
            "Type": "AUTOMATION",
            "Description": "Automation task to disable read/write access on
 public S3 buckets",
            "MaxConcurrency": "10",
            "MaxErrors": "5",
            "Name": "My-Disable-S3-Public-Read-Write-Access-Automation-Task",
            "Priority": 0,
            "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
            "Targets": [
                {
                    "Key": "WindowTargetIds",
                    "Values": [
                        "da89dcc3-7f9c-481d-ba2b-edcb7EXAMPLE"
                    ]
                }
            ]
        }
    ]
}
```

Exemple 2 : Pour répertorier toutes les tâches d'une fenêtre de maintenance qui invoque le document de RunPowerShellScript commande AWS-

L'`describe-maintenance-window-tasks`exemple suivant répertorie toutes les tâches pour la fenêtre de maintenance spécifiée qui appelle le document de `AWS-RunPowerShellScript` commande.

```
aws ssm describe-maintenance-window-tasks \
    --window-id "mw-ab12cd34eEXAMPLE" \
    --filters "Key=TaskArn,Values=AWS-RunPowerShellScript"
```

Sortie :

```
{
    "Tasks": [
        {
            "WindowId": "mw-ab12cd34eEXAMPLE",
            "WindowTaskId": "0d36e6b4-3a4f-411e-adcb-3558eEXAMPLE",
            "TaskArn": "AWS-RunPowerShellScript",
            "Type": "RUN_COMMAND",
            "Targets": [
                {
                    "Key": "WindowTargetIds",
                    "Values": [
                        "da89dcc3-7f9c-481d-ba2b-edcb7EXAMPLE"
                    ]
                }
            ],
            "TaskParameters": {},
            "Priority": 1,
            "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
            "MaxConcurrency": "1",
            "MaxErrors": "1",
            "Name": "MyTask"
        }
    ]
}
```

Exemple 3 : pour répertorier toutes les tâches d'une fenêtre de maintenance ayant une priorité de 3

L'`describe-maintenance-window-tasks`exemple suivant répertorie toutes les tâches correspondant à la fenêtre de maintenance spécifiée qui ont une valeur `Priority` de`3`.

```
aws ssm describe-maintenance-window-tasks \
    --window-id "mw-ab12cd34eEXAMPLE" \
    --filters "Key=Priority,Values=3"
```

Sortie :

```
{
    "Tasks": [
        {
            "WindowId": "mw-ab12cd34eEXAMPLE",
```

```
            "WindowTaskId": "0d36e6b4-3a4f-411e-adcb-3558eEXAMPLE",
            "TaskArn": "AWS-RunPowerShellScript",
            "Type": "RUN_COMMAND",
            "Targets": [
                {
                    "Key": "WindowTargetIds",
                    "Values": [
                        "da89dcc3-7f9c-481d-ba2b-edcb7EXAMPLE"
                    ]
                }
            ],
            "TaskParameters": {},
            "Priority": 3,
            "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
            "MaxConcurrency": "1",
            "MaxErrors": "1",
            "Name": "MyRunCommandTask"
        },
        {
            "WindowId": "mw-ab12cd34eEXAMPLE",
            "WindowTaskId": "ee45feff-ad65-4a6c-b478-5cab8EXAMPLE",
            "TaskArn": "AWS-RestartEC2Instance",
            "Type": "AUTOMATION",
            "Targets": [
                {
                    "Key": "WindowTargetIds",
                    "Values": [
                        "da89dcc3-7f9c-481d-ba2b-edcb7EXAMPLE"
                    ]
                }
            ],
            "TaskParameters": {},
            "Priority": 3,
            "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
            "MaxConcurrency": "10",
            "MaxErrors": "5",
            "Name": "My-Automation-Task",
            "Description": "A description for my Automation task"
        }
    ]
}
```

Exemple 4 : pour répertorier toutes les tâches d'une fenêtre de maintenance dont la priorité est égale à 1 et pour utiliser la commande Exécuter

Cet `describe-maintenance-window-tasks` exemple répertorie toutes les tâches correspondant à la fenêtre de maintenance spécifiée qui ont une valeur `Priority` de 1 et une utilisation`Run Command`.

```
aws ssm describe-maintenance-window-tasks \
    --window-id "mw-ab12cd34eEXAMPLE" \
    --filters "Key=Priority,Values=1" "Key=TaskType,Values=RUN_COMMAND"
```

Sortie :

```
{
    "Tasks": [
        {
            "WindowId": "mw-ab12cd34eEXAMPLE",
            "WindowTaskId": "0d36e6b4-3a4f-411e-adcb-3558eEXAMPLE",
            "TaskArn": "AWS-RunPowerShellScript",
            "Type": "RUN_COMMAND",
            "Targets": [
                {
                    "Key": "WindowTargetIds",
                    "Values": [
                        "da89dcc3-7f9c-481d-ba2b-edcb7EXAMPLE"
                    ]
                }
            ],
            "TaskParameters": {},
            "Priority": 1,
            "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
            "MaxConcurrency": "1",
            "MaxErrors": "1",
            "Name": "MyRunCommandTask"
        }
    ]
}
```

Pour plus d'informations, consultez la section Afficher les informations sur les fenêtres de maintenance (AWS CLI) dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous DescribeMaintenanceWindowTasksà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple répertorie toutes les tâches d'une fenêtre de maintenance.

```
Get-SSMMaintenanceWindowTaskList -WindowId "mw-06cf17cbefcb4bf4f"
```

Sortie :

```
LoggingInfo    :
MaxConcurrency : 1
MaxErrors      : 1
Priority       : 10
ServiceRoleArn : arn:aws:iam::123456789012:role/MaintenanceWindowsRole
Targets        : {InstanceIds}
TaskArn        : AWS-RunShellScript
TaskParameters : {[commands,
 Amazon.SimpleSystemsManagement.Model.MaintenanceWindowTaskParameterValueExpression]}
Type           : RUN_COMMAND
WindowId       : mw-06cf17cbefcb4bf4f
WindowTaskId   : a23e338d-ff30-4398-8aa3-09cd052ebf17
```

- Pour plus de détails sur l'API, reportez-vous DescribeMaintenanceWindowTasksà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeMaintenanceWindows`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribeMaintenanceWindows`.

CLI

AWS CLI

Exemple 1 : pour répertorier toutes les fenêtres de maintenance

L'`describe-maintenance-windows`exemple suivant répertorie toutes les fenêtres de maintenance de votre AWS compte dans la région actuelle.

```
aws ssm describe-maintenance-windows
```

Sortie :

```
{
    "WindowIdentities": [
        {
            "WindowId": "mw-0ecb1226ddEXAMPLE",
            "Name": "MyMaintenanceWindow-1",
            "Enabled": true,
            "Duration": 2,
            "Cutoff": 1,
            "Schedule": "rate(180 minutes)",
            "NextExecutionTime": "2020-02-12T23:19:20.596Z"
        },
        {
            "WindowId": "mw-03eb9db428EXAMPLE",
            "Name": "MyMaintenanceWindow-2",
            "Enabled": true,
            "Duration": 3,
            "Cutoff": 1,
            "Schedule": "rate(7 days)",
            "NextExecutionTime": "2020-02-17T23:22:00.956Z"
        },
    ]
}
```

Exemple 2 : pour répertorier toutes les fenêtres de maintenance activées

L'`describe-maintenance-windows`exemple suivant répertorie toutes les fenêtres de maintenance activées.

```
aws ssm describe-maintenance-windows \
```

```
    --filters "Key=Enabled,Values=true"
```

Exemple 3 : pour répertorier les fenêtres de maintenance correspondant à un nom spécifique

Cet `describe-maintenance-windows` exemple répertorie toutes les fenêtres de maintenance portant le nom spécifié.

```
aws ssm describe-maintenance-windows \
    --filters "Key=Name,Values=MyMaintenanceWindow"
```

Pour plus d'informations, consultez la section Afficher les informations relatives à la maintenance Windows (AWS CLI) dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, consultez DescribeMaintenanceWindows dans AWS CLI Command Reference.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple répertorie toutes les fenêtres de maintenance de votre compte.

```
Get-SSMMaintenanceWindowList
```

Sortie :

```
Cutoff   : 1
Duration : 4
Enabled  : True
Name     : My-First-Maintenance-Window
WindowId : mw-06d59c1a07c022145
```

- Pour plus de détails sur l'API, consultez DescribeMaintenanceWindows dans le manuel de référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **DescribeOpsItems** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribeOpsItems`.

CLI

AWS CLI

Pour répertorier un ensemble de OpsItems

L'`describe-ops-items`exemple suivant affiche une liste de toutes les offres ouvertes OpsItems dans votre AWS compte.

```
aws ssm describe-ops-items \
    --ops-item-filters "Key=Status,Values=Open,Operator=Equal"
```

Sortie :

```
{
    "OpsItemSummaries": [
        {
            "CreatedBy": "arn:aws:sts::111222333444:assumed-role/OpsItem-CWE-
Role/fbf77cbe264a33509569f23e4EXAMPLE",
            "CreatedTime": "2020-03-14T17:02:46.375000-07:00",
            "LastModifiedBy": "arn:aws:sts::111222333444:assumed-role/OpsItem-
CWE-Role/fbf77cbe264a33509569f23e4EXAMPLE",
            "LastModifiedTime": "2020-03-14T17:02:46.375000-07:00",
            "Source": "SSM",
            "Status": "Open",
            "OpsItemId": "oi-7cfc5EXAMPLE",
            "Title": "SSM Maintenance Window execution failed",
            "OperationalData": {
                "/aws/dedup": {
                    "Value": "{\"dedupString\":\"SSMOpsItems-SSM-maintenance-
window-execution-failed\"}",
                    "Type": "SearchableString"
                },
                "/aws/resources": {
                    "Value": "[{\"arn\":\"arn:aws:ssm:us-
east-2:111222333444:maintenancewindow/mw-034093d322EXAMPLE\"}]",
                    "Type": "SearchableString"
                }
```

```
            },
            "Category": "Availability",
            "Severity": "3"
        },
        {
            "CreatedBy": "arn:aws:sts::1112223233444:assumed-role/OpsItem-CWE-
Role/fbf77cbe264a33509569f23e4EXAMPLE",
            "CreatedTime": "2020-02-26T11:43:15.426000-08:00",
            "LastModifiedBy": "arn:aws:sts::111222333444:assumed-role/OpsItem-
CWE-Role/fbf77cbe264a33509569f23e4EXAMPLE",
            "LastModifiedTime": "2020-02-26T11:43:15.426000-08:00",
            "Source": "EC2",
            "Status": "Open",
            "OpsItemId": "oi-6f966EXAMPLE",
            "Title": "EC2 instance stopped",
            "OperationalData": {
                "/aws/automations": {
                    "Value": "[ { \"automationType\": \"AWS:SSM:Automation\",
 \"automationId\": \"AWS-RestartEC2Instance\" } ]",
                    "Type": "SearchableString"
                },
                "/aws/dedup": {
                    "Value": "{\"dedupString\":\"SSMOpsItems-EC2-instance-stopped
\"}",
                    "Type": "SearchableString"
                },
                "/aws/resources": {
                    "Value": "[{\"arn\":\"arn:aws:ec2:us-
east-2:111222333444:instance/i-0beccfbc02EXAMPLE\"}]",
                    "Type": "SearchableString"
                }
            },
            "Category": "Availability",
            "Severity": "3"
        }
    ]
}
```

Pour plus d'informations, reportez-vous à la section Travailler avec OpsItems dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, consultez la section DescribeOpsÉléments du manuel de référence des AWS CLI commandes.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
   public static void describeOpsItems(SsmClient ssmClient, String key) {
       try {
           OpsItemFilter filter = OpsItemFilter.builder()
               .key(OpsItemFilterKey.OPS_ITEM_ID)
               .values(key)
               .operator(OpsItemFilterOperator.EQUAL)
               .build();

           DescribeOpsItemsRequest itemsRequest =
DescribeOpsItemsRequest.builder()
               .maxResults(10)
               .opsItemFilters(filter)
               .build();

           DescribeOpsItemsResponse itemsResponse =
ssmClient.describeOpsItems(itemsRequest);
           List<OpsItemSummary> items = itemsResponse.opsItemSummaries();
           for (OpsItemSummary item : items) {
               System.out.println("The item title is " + item.title() +" and the
status is "+item.status().toString());
           }

       } catch (SsmException e) {
           System.err.println(e.getMessage());
           System.exit(1);
       }
   }
```

- Pour plus de détails sur l'API, consultez la section DescribeOpsÉléments du manuel de référence des AWS SDK for Java 2.x API.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeParameters`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribeParameters`.

CLI

AWS CLI

Exemple 1 : pour répertorier tous les paramètres

L'`describe-parameters`exemple suivant répertorie tous les paramètres du AWS compte courant et de la région.

```
aws ssm describe-parameters
```

Sortie :

```
{
    "Parameters": [
        {
            "Name": "MySecureStringParameter",
            "Type": "SecureString",
            "KeyId": "alias/aws/ssm",
            "LastModifiedDate": 1582155479.205,
            "LastModifiedUser": "arn:aws:sts::111222333444:assumed-role/Admin/
Richard-Roe-Managed",
            "Description": "This is a SecureString parameter",
            "Version": 2,
            "Tier": "Advanced",
            "Policies": [
                {
                    "PolicyText": "{\"Type\":\"Expiration\",\"Version\":\"1.0\",
\"Attributes\":{\"Timestamp\":\"2020-07-07T22:30:00Z\"}}",
                    "PolicyType": "Expiration",
                    "PolicyStatus": "Pending"
                },
                {
                    "PolicyText": "{\"Type\":\"ExpirationNotification\",\"Version
\":\"1.0\",\"Attributes\":{\"Before\":\"12\",\"Unit\":\"Hours\"}}",
```

```
                    "PolicyType": "ExpirationNotification",
                    "PolicyStatus": "Pending"
                }
            ]
        },
        {
            "Name": "MyStringListParameter",
            "Type": "StringList",
            "LastModifiedDate": 1582154764.222,
            "LastModifiedUser": "arn:aws:iam::111222333444:user/Mary-Major",
            "Description": "This is a StringList parameter",
            "Version": 1,
            "Tier": "Standard",
            "Policies": []
        },
        {
            "Name": "MyStringParameter",
            "Type": "String",
            "LastModifiedDate": 1582154711.976,
            "LastModifiedUser": "arn:aws:iam::111222333444:user/Alejandro-
Rosalez",
            "Description": "This is a String parameter",
            "Version": 1,
            "Tier": "Standard",
            "Policies": []
        },
        {
            "Name": "latestAmi",
            "Type": "String",
            "LastModifiedDate": 1580862415.521,
            "LastModifiedUser": "arn:aws:sts::111222333444:assumed-role/lambda-
ssm-role/Automation-UpdateSSM-Param",
            "Version": 3,
            "Tier": "Standard",
            "Policies": []
        }
    ]
}
```

Exemple 2 : pour répertorier tous les paramètres correspondant à des métadonnées spécifiques

Cet `describe-parameters` exemple répertorie tous les paramètres correspondant à un filtre.

aws ssm describe-parameters --filters « Clé = type, valeurs = » StringList

Sortie :

```
{
    "Parameters": [
        {
            "Name": "MyStringListParameter",
            "Type": "StringList",
            "LastModifiedDate": 1582154764.222,
            "LastModifiedUser": "arn:aws:iam::111222333444:user/Mary-Major",
            "Description": "This is a StringList parameter",
            "Version": 1,
            "Tier": "Standard",
            "Policies": []
        }
    ]
}
```

Pour plus d'informations, reportez-vous à la section Searching for Systems Manager Parameters dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous DescribeParametersà la section Référence des AWS CLI commandes.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.ssm.SsmClient;
import software.amazon.awssdk.services.ssm.model.GetParameterRequest;
```

```
import software.amazon.awssdk.services.ssm.model.GetParameterResponse;
import software.amazon.awssdk.services.ssm.model.SsmException;

/**
 * Before running this Java V2 code example, set up your development
 * environment, including your credentials.
 *
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-
started.html
 */
public class GetParameter {
    public static void main(String[] args) {
        final String usage = """

                Usage:
                    <paraName>

                Where:
                    paraName - The name of the parameter.
                """;

        if (args.length != 1) {
            System.out.println(usage);
            System.exit(1);
        }

        String paraName = args[0];
        Region region = Region.US_EAST_1;
        SsmClient ssmClient = SsmClient.builder()
                .region(region)
                .build();

        getParaValue(ssmClient, paraName);
        ssmClient.close();
    }

    public static void getParaValue(SsmClient ssmClient, String paraName) {
        try {
            GetParameterRequest parameterRequest = GetParameterRequest.builder()
                    .name(paraName)
                    .build();
```

```
            GetParameterResponse parameterResponse =
 ssmClient.getParameter(parameterRequest);
            System.out.println("The parameter value is " +
 parameterResponse.parameter().value());

        } catch (SsmException e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }
}
```

- Pour plus de détails sur l'API, reportez-vous DescribeParametersà la section Référence des AWS SDK for Java 2.x API.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple répertorie tous les paramètres.

```
Get-SSMParameterList
```

Sortie :

```
Description      :
KeyId            :
LastModifiedDate : 3/3/2017 6:58:23 PM
LastModifiedUser : arn:aws:iam::123456789012:user/admin
Name             : Welcome
Type             : String
```

- Pour plus de détails sur l'API, reportez-vous DescribeParametersà la section Référence des AWS Tools for PowerShell applets de commande.

Rust

SDK pour Rust

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
async fn show_parameters(client: &Client) -> Result<(), Error> {
    let resp = client.describe_parameters().send().await?;

    for param in resp.parameters() {
        println!("  {}", param.name().unwrap_or_default());
    }

    Ok(())
}
```

- Pour plus de détails sur l'API, voir DescribeParameterslа section de référence de l'API AWS SDK for Rust.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribePatchBaselines`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribePatchBaselines`.

CLI

AWS CLI

Exemple 1 : pour répertorier toutes les lignes de base des correctifs

L'`describe-patch-baselines`exemple suivant permet de récupérer les détails de toutes les lignes de base des correctifs de votre compte dans la région actuelle.

```
aws ssm describe-patch-baselines
```

Sortie :

```
{
    "BaselineIdentities": [
        {
            "BaselineName": "AWS-SuseDefaultPatchBaseline",
            "DefaultBaseline": true,
            "BaselineDescription": "Default Patch Baseline for Suse Provided by
 AWS.",
            "BaselineId": "arn:aws:ssm:us-east-2:733109147000:patchbaseline/
pb-0123fdb36e334a3b2",
            "OperatingSystem": "SUSE"
        },
        {
            "BaselineName": "AWS-DefaultPatchBaseline",
            "DefaultBaseline": false,
            "BaselineDescription": "Default Patch Baseline Provided by AWS.",
            "BaselineId": "arn:aws:ssm:us-east-2:733109147000:patchbaseline/
pb-020d361a05defe4ed",
            "OperatingSystem": "WINDOWS"
        },
        ...
        {
            "BaselineName": "MyWindowsPatchBaseline",
            "DefaultBaseline": true,
            "BaselineDescription": "My patch baseline for EC2 instances for
 Windows Server",
            "BaselineId": "pb-0ad00e0dd7EXAMPLE",
            "OperatingSystem": "WINDOWS"
        }
    ]
}
```

Exemple 2 : Pour répertorier toutes les lignes de base de correctifs fournies par AWS

L'`describe-patch-baselines`exemple suivant répertorie toutes les lignes de base de correctifs fournies par AWS.

```
aws ssm describe-patch-baselines \
    --filters "Key=OWNER,Values=[AWS]"
```

Exemple 3 : pour répertorier toutes les lignes de base de correctifs que vous possédez

L'`describe-patch-baselines`exemple suivant répertorie toutes les lignes de base de correctifs personnalisées créées dans votre compte dans la région actuelle.

```
aws ssm describe-patch-baselines \
    --filters "Key=OWNER,Values=[Self]"
```

Pour plus d'informations, reportez-vous à la section À propos des lignes de base de correctifs prédéfinies et personnalisées dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, consultez la section DescribePatchLignes de base dans AWS CLI Command Reference.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple répertorie toutes les lignes de base des correctifs.

```
Get-SSMPatchBaseline
```

Sortie :

```
BaselineDescription                                                BaselineId
                                                  BaselineName
------------------                                                ----------
                                                  ------------
Default Patch Baseline Provided by AWS.                           arn:aws:ssm:us-
west-2:123456789012:patchbaseline/pb-04fb4ae6142167966 AWS-DefaultP...
Baseline containing all updates approved for production systems
 pb-045f10b4f382baeda
 Production-B...
Baseline containing all updates approved for production systems
 pb-0a2f1059b670ebd31
 Production-B...
```

Exemple 2 : Cet exemple répertorie toutes les lignes de base de correctifs fournies par AWS. La syntaxe utilisée dans cet exemple nécessite PowerShell la version 3 ou ultérieure.

```
$filter1 = @{Key="OWNER";Values=@("AWS")}
```

Sortie :

```
Get-SSMPatchBaseline -Filter $filter1
```

Exemple 3 : Cet exemple répertorie toutes les lignes de base des correctifs dont vous êtes le propriétaire. La syntaxe utilisée dans cet exemple nécessite PowerShell la version 3 ou ultérieure.

```
$filter1 = @{Key="OWNER";Values=@("Self")}
```

Sortie :

```
Get-SSMPatchBaseline -Filter $filter1
```

Exemple 4 : Avec PowerShell la version 2, vous devez utiliser New-Object pour créer chaque balise.

```
$filter1 = New-Object
 Amazon.SimpleSystemsManagement.Model.PatchOrchestratorFilter
$filter1.Key = "OWNER"
$filter1.Values = "AWS"

Get-SSMPatchBaseline -Filter $filter1
```

Sortie :

```
BaselineDescription                     BaselineId
                            BaselineName            DefaultBaselin

                                                    e
------------------                      ----------
                            ------------            --------------
Default Patch Baseline Provided by AWS. arn:aws:ssm:us-
west-2:123456789012:patchbaseline/pb-04fb4ae6142167966 AWS-DefaultPatchBaseline
 True
```

- Pour plus de détails sur l'API, consultez la section DescribePatchLignes de base dans la référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribePatchGroupState`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribePatchGroupState`.

CLI

AWS CLI

Pour obtenir l'état d'un groupe de correctifs

L'`describe-patch-group-state`exemple suivant extrait le résumé de haut niveau de conformité des correctifs pour un groupe de correctifs.

```
aws ssm describe-patch-group-state \
    --patch-group "Production"
```

Sortie :

```
{
    "Instances": 21,
    "InstancesWithCriticalNonCompliantPatches": 1,
    "InstancesWithFailedPatches": 2,
    "InstancesWithInstalledOtherPatches": 3,
    "InstancesWithInstalledPatches": 21,
    "InstancesWithInstalledPendingRebootPatches": 2,
    "InstancesWithInstalledRejectedPatches": 1,
    "InstancesWithMissingPatches": 3,
    "InstancesWithNotApplicablePatches": 4,
    "InstancesWithOtherNonCompliantPatches": 1,
    "InstancesWithSecurityNonCompliantPatches": 1,
    "InstancesWithUnreportedNotApplicablePatches": 2
}
```

Pour plus d'informations, consultez About patch groups__ < https://docs.aws.amazon.com/systems-manager/latest/userguide/sysman-patch-patchgroups.html> et Understanding patch compliance values dans le guide de l'utilisateur de Systems AWS Manager.

- Pour plus de détails sur l'API, reportez-vous DescribePatchGroupStateà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple permet d'obtenir le résumé de haut niveau de conformité des correctifs pour un groupe de correctifs.

```
Get-SSMPatchGroupState -PatchGroup "Production"
```

Sortie :

```
Instances                          : 4
InstancesWithFailedPatches         : 1
InstancesWithInstalledOtherPatches : 4
InstancesWithInstalledPatches      : 3
InstancesWithMissingPatches        : 0
InstancesWithNotApplicablePatches  : 0
```

- Pour plus de détails sur l'API, reportez-vous DescribePatchGroupStateà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribePatchGroups`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`DescribePatchGroups`.

CLI

AWS CLI

Pour afficher les enregistrements de groupes de correctifs

L'`describe-patch-groups`exemple suivant répertorie les enregistrements de groupes de correctifs.

```
aws ssm describe-patch-groups
```

Sortie :

```
{
    "Mappings": [
        {
            "PatchGroup": "Production",
            "BaselineIdentity": {
                "BaselineId": "pb-0123456789abcdef0",
                "BaselineName": "ProdPatching",
                "OperatingSystem": "WINDOWS",
                "BaselineDescription": "Patches for Production",
                "DefaultBaseline": false
            }
        },
        {
            "PatchGroup": "Development",
            "BaselineIdentity": {
                "BaselineId": "pb-0713accee01234567",
                "BaselineName": "DevPatching",
                "OperatingSystem": "WINDOWS",
                "BaselineDescription": "Patches for Development",
                "DefaultBaseline": true
            }
        },
        ...
    ]
}
```

Pour plus d'informations, consultez les sections Create a Patch Group__ < https://docs.aws.amazon.com/systems-manager/latest/userguide/sysman-patch-group-tagging.html> et Add a Patch Group to a Patch Baseline dans le Guide de l'utilisateur de Systems AWS Manager.

- Pour plus de détails sur l'API, consultez la section DescribePatchGroupes dans AWS CLI la référence des commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple répertorie les enregistrements de groupes de correctifs.

```
Get-SSMPatchGroup
```

Sortie :

```
BaselineIdentity                                          PatchGroup
----------------                                          ----------
Amazon.SimpleSystemsManagement.Model.PatchBaselineIdentity Production
```

- Pour plus de détails sur l'API, consultez la section DescribePatchGroups in AWS Tools for PowerShell Cmdlet Reference.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`GetAutomationExecution`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`GetAutomationExecution`.

CLI

AWS CLI

Pour afficher les détails d'une exécution automatisée

L'`get-automation-execution`exemple suivant affiche des informations détaillées sur l'exécution d'une automatisation.

```
aws ssm get-automation-execution \
    --automation-execution-id 73c8eef8-f4ee-4a05-820c-e354fEXAMPLE
```

Sortie :

```
{
    "AutomationExecution": {
```

```
        "AutomationExecutionId": "73c8eef8-f4ee-4a05-820c-e354fEXAMPLE",
        "DocumentName": "AWS-StartEC2Instance",
        "DocumentVersion": "1",
        "ExecutionStartTime": 1583737233.748,
        "ExecutionEndTime": 1583737234.719,
        "AutomationExecutionStatus": "Success",
        "StepExecutions": [
            {
                "StepName": "startInstances",
                "Action": "aws:changeInstanceState",
                "ExecutionStartTime": 1583737234.134,
                "ExecutionEndTime": 1583737234.672,
                "StepStatus": "Success",
                "Inputs": {
                    "DesiredState": "\"running\"",
                    "InstanceIds": "[\"i-0cb99161f6EXAMPLE\"]"
                },
                "Outputs": {
                    "InstanceStates": [
                        "running"
                    ]
                },
                "StepExecutionId": "95e70479-cf20-4d80-8018-7e4e2EXAMPLE",
                "OverriddenParameters": {}
            }
        ],
        "StepExecutionsTruncated": false,
        "Parameters": {
            "AutomationAssumeRole": [
                ""
            ],
            "InstanceId": [
                "i-0cb99161f6EXAMPLE"
            ]
        },
        "Outputs": {},
        "Mode": "Auto",
        "ExecutedBy": "arn:aws:sts::29884EXAMPLE:assumed-role/mw_service_role/
OrchestrationService",
        "Targets": [],
        "ResolvedTargets": {
            "ParameterValues": [],
            "Truncated": false
        }
```

```
    }
}
```

Pour plus d'informations, reportez-vous à la section Procédure pas à pas : patcher une AMI Linux (AWS CLI) dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, consultez la section GetAutomationExécution dans AWS CLI la référence des commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple affiche les détails d'une exécution d'automatisation.

```
Get-SSMAutomationExecution -AutomationExecutionId "4105a4fc-
f944-11e6-9d32-8fb2db27a909"
```

Sortie :

```
AutomationExecutionId     : 4105a4fc-f944-11e6-9d32-8fb2db27a909
AutomationExecutionStatus : Failed
DocumentName              : AWS-UpdateLinuxAmi
DocumentVersion           : 1
ExecutionEndTime          : 2/22/2017 9:17:08 PM
ExecutionStartTime        : 2/22/2017 9:17:02 PM
FailureMessage            : Step launchInstance failed maximum allowed times. You
 are not authorized to perform this operation. Encoded
                            authorization failure message:
 B_V2QyyN7NhSZQYpmVzpEc4oSnj2GLTNYnXUHsTbqJkNMoDgubmbtthLmZyaiUYekORIrA42-
fv1x-04q5Fjff6glh
                            Yb6TI5b0GQeeNrpwNvpDzmO-
PSR1swlAbg9fdM9BcNjyrznspUkWpuKu9EC1Ou6v3OXU1KC9nZ7mPlWMFZNkSioQqpwWEvMw-
GZktsQzm67qOhUhBNOLWYhbS
                            pkfiqzY-5nw3S0obx30fhd3EJa5O_-
GjV_a0nFXQJa70ik40bFOrEh3MtCSbrQT6--DvFy_FQ8TKvkIXadyVskeJI84XOF5WmA60f1pi5GI08i-
nRfZS6oDeU

 gELBjjoFKD8s3L2aI0B6umWVxnQOjqhQRxwJ53b54sZJ2PW3v_mtg9-q0CK0ezS3xfh_y0ilaUGOAZG-
xjQFuvU_JZedWpla3xi-MZsmblAifBI
                            (Service: AmazonEC2; Status Code: 403; Error Code:
 UnauthorizedOperation; Request ID:
```

```
                            6a002f94-ba37-43fd-99e6-39517715fce5)
Outputs                   : {[createImage.ImageId,
 Amazon.Runtime.Internal.Util.AlwaysSendList`1[System.String]]}
Parameters                : {[AutomationAssumeRole,
 Amazon.Runtime.Internal.Util.AlwaysSendList`1[System.String]], [InstanceIamRole,

 Amazon.Runtime.Internal.Util.AlwaysSendList`1[System.String]], [SourceAmiId,

 Amazon.Runtime.Internal.Util.AlwaysSendList`1[System.String]]}
StepExecutions            : {launchInstance, updateOSSoftware, stopInstance,
 createImage...}
```

Exemple 2 : Cet exemple répertorie les détails des étapes pour l'identifiant d'exécution automatique donné

```
Get-SSMAutomationExecution -AutomationExecutionId e1d2bad3-4567-8901-
ae23-456c7c8901be | Select-Object -ExpandProperty StepExecutions | Select-Object
 StepName, Action, StepStatus, ValidNextSteps
```

Sortie :

```
StepName                  Action                  StepStatus ValidNextSteps
--------                  ------                  ---------- --------------
LaunchInstance            aws:runInstances        Success
 {OSCompatibilityCheck}
OSCompatibilityCheck      aws:runCommand          Success    {RunPreUpdateScript}
RunPreUpdateScript        aws:runCommand          Success    {UpdateEC2Config}
UpdateEC2Config           aws:runCommand          Cancelled  {}
UpdateSSMAgent            aws:runCommand          Pending    {}
UpdateAWSPVDriver         aws:runCommand          Pending    {}
UpdateAWSEnaNetworkDriver aws:runCommand          Pending    {}
UpdateAWSNVMe             aws:runCommand          Pending    {}
InstallWindowsUpdates     aws:runCommand          Pending    {}
RunPostUpdateScript       aws:runCommand          Pending    {}
RunSysprepGeneralize      aws:runCommand          Pending    {}
StopInstance              aws:changeInstanceState Pending    {}
CreateImage               aws:createImage         Pending    {}
TerminateInstance         aws:changeInstanceState Pending    {}
```

- Pour plus de détails sur l'API, consultez la section GetAutomationExecution in AWS Tools for PowerShell Cmdlet Reference.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`GetCommandInvocation`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`GetCommandInvocation`.

CLI

AWS CLI

Pour afficher les détails d'un appel de commande

L'`get-command-invocation`exemple suivant répertorie toutes les invocations de la commande spécifiée sur l'instance spécifiée.

```
aws ssm get-command-invocation \
    --command-id "ef7fdfd8-9b57-4151-a15c-db9a12345678" \
    --instance-id "i-1234567890abcdef0"
```

Sortie :

```
{
    "CommandId": "ef7fdfd8-9b57-4151-a15c-db9a12345678",
    "InstanceId": "i-1234567890abcdef0",
    "Comment": "b48291dd-ba76-43e0-b9df-13e11ddaac26:6960febb-2907-4b59-8e1a-
d6ce8EXAMPLE",
    "DocumentName": "AWS-UpdateSSMAgent",
    "DocumentVersion": "",
    "PluginName": "aws:updateSsmAgent",
    "ResponseCode": 0,
    "ExecutionStartDateTime": "2020-02-19T18:18:03.419Z",
    "ExecutionElapsedTime": "PT0.091S",
    "ExecutionEndDateTime": "2020-02-19T18:18:03.419Z",
    "Status": "Success",
    "StatusDetails": "Success",
    "StandardOutputContent": "Updating amazon-ssm-agent from 2.3.842.0 to latest
\nSuccessfully downloaded https://s3.us-east-2.amazonaws.com/amazon-ssm-us-
east-2/ssm-agent-manifest.json\namazon-ssm-agent 2.3.842.0 has already been
 installed, update skipped\n",
    "StandardOutputUrl": "",
    "StandardErrorContent": "",
```

```
    "StandardErrorUrl": "",
    "CloudWatchOutputConfig": {
        "CloudWatchLogGroupName": "",
        "CloudWatchOutputEnabled": false
    }
}
```

Pour plus d'informations, consultez la section Understanding Command Statuses dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, voir GetCommandInvocation dans la référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple affiche les détails d'une commande exécutée sur une instance.

```
Get-SSMCommandInvocationDetail -InstanceId "i-0cb2b964d3e14fd9f" -CommandId
 "b8eac879-0541-439d-94ec-47a80d554f44"
```

Sortie :

```
CommandId              : b8eac879-0541-439d-94ec-47a80d554f44
Comment                : IP config
DocumentName           : AWS-RunShellScript
ExecutionElapsedTime   : PT0.004S
ExecutionEndDateTime   : 2017-02-22T20:13:16.651Z
ExecutionStartDateTime : 2017-02-22T20:13:16.651Z
InstanceId             : i-0cb2b964d3e14fd9f
PluginName             : aws:runShellScript
ResponseCode           : 0
StandardErrorContent   :
StandardErrorUrl       :
StandardOutputContent  :
StandardOutputUrl      :
Status                 : Success
StatusDetails          : Success
```

- Pour plus de détails sur l'API, consultez la section GetCommandInvocation dans la référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`GetConnectionStatus`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`GetConnectionStatus`.

CLI

AWS CLI

Pour afficher l'état de connexion d'une instance gérée

Cet `get-connection-status` exemple renvoie l'état de connexion de l'instance gérée spécifiée.

```
aws ssm get-connection-status \
    --target i-1234567890abcdef0
```

Sortie :

```
{
    "Target": "i-1234567890abcdef0",
    "Status": "connected"
}
```

- Pour plus de détails sur l'API, consultez la section GetConnectionÉtat dans AWS CLI la référence des commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple récupère l'état de connexion au gestionnaire de session d'une instance afin de déterminer si elle est connectée et prête à recevoir des connexions au gestionnaire de session.

```
Get-SSMConnectionStatus -Target i-0a1caf234f12d3dc4
```

Sortie :

```
Status    Target
------    ------
Connected i-0a1caf234f12d3dc4
```

- Pour plus de détails sur l'API, consultez la section GetConnectionStatus in AWS Tools for PowerShell Cmdlet Reference.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **GetDefaultPatchBaseline** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliserGetDefaultPatchBaseline.

CLI

AWS CLI

Exemple 1 : pour afficher la ligne de base des correctifs Windows par défaut

L'get-default-patch-baselineexemple suivant permet de récupérer les détails de la ligne de base de correctifs par défaut pour Windows Server.

```
aws ssm get-default-patch-baseline
```

Sortie :

```
{
  "BaselineId": "pb-0713accee01612345",
  "OperatingSystem": "WINDOWS"
}
```

Exemple 2 : pour afficher la ligne de base de correctifs par défaut pour Amazon Linux

L'get-default-patch-baselineexemple suivant récupère les détails de la ligne de base de correctifs par défaut pour Amazon Linux.

```
aws ssm get-default-patch-baseline \
    --operating-system AMAZON_LINUX
```

Sortie :

```
{
    "BaselineId": "pb-047c6eb9c8fc12345",
    "OperatingSystem": "AMAZON_LINUX"
}
```

Pour plus d'informations, voir À propos des lignes de base de correctifs prédéfinies et personnalisées __ < https://docs.aws.amazon.com/systems-manager/latest/userguide/sysman-patch-baselines.html> et Définir une ligne de base de correctifs existante par défaut dans le guide de l'utilisateur de Systems AWS Manager.

- Pour plus de détails sur l'API, reportez-vous GetDefaultPatchBaselineà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple affiche la ligne de base de correctifs par défaut.

```
Get-SSMDefaultPatchBaseline
```

Sortie :

```
arn:aws:ssm:us-west-2:123456789012:patchbaseline/pb-04fb4ae6142167966
```

- Pour plus de détails sur l'API, reportez-vous GetDefaultPatchBaselineà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`GetDeployablePatchSnapshotForInstance`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`GetDeployablePatchSnapshotForInstance`.

CLI

AWS CLI

Pour récupérer l'instantané actuel de la ligne de base du correctif, une instance utilise

L'`get-deployable-patch-snapshot-for-instance`exemple suivant récupère les détails du cliché actuel pour la ligne de base de correctif spécifiée utilisée par une instance. Cette commande doit être exécutée depuis l'instance à l'aide des informations d'identification de l'instance. Pour vous assurer qu'il utilise les informations d'identification de l'instance, exécutez `aws configure` et spécifiez uniquement la région de votre instance. Laissez les `Secret Key` champs `Access Key` et vides.

Conseil : `uuidgen` Utilisez-le pour générer un`snapshot-id`.

```
aws ssm get-deployable-patch-snapshot-for-instance \
    --instance-id "i-1234567890abcdef0" \
    --snapshot-id "521c3536-930c-4aa9-950e-01234567abcd"
```

Sortie :

```
{
    "InstanceId": "i-1234567890abcdef0",
    "SnapshotId": "521c3536-930c-4aa9-950e-01234567abcd",
    "Product": "AmazonLinux2018.03",
    "SnapshotDownloadUrl": "https://patch-baseline-snapshot-us-
east-1.s3.amazonaws.com/
ed85194ef27214f5984f28b4d664d14f7313568fea7d4b6ac6c10ad1f729d7e7-773304212436/
AMAZON_LINUX-521c3536-930c-4aa9-950e-01234567abcd?X-Amz-
Algorithm=AWS4-HMAC-SHA256&X-Amz-Date=20190215T164031Z&X-Amz-
SignedHeaders=host&X-Amz-Expires=86400&X-Amz-Credential=AKIAJ5C56P35AEBRX2QQ
%2F20190215%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-
Signature=efaaaf6e3878e77f48a6697e015efdbda9c426b09c5822055075c062f6ad2149"
}
```

Pour plus d'informations, reportez-vous à la section Nom du paramètre : Snapshot ID dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous GetDeployablePatchSnapshotForInstanceà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple affiche l'instantané actuel de la ligne de base de correctif utilisée par une instance. Cette commande doit être exécutée depuis l'instance à l'aide des informations d'identification de l'instance. Pour s'assurer qu'il utilise les informations d'identification de l'instance, l'exemple transmet un **`Amazon.Runtime.InstanceProfileAWSCredentials`** objet au paramètre Credentials.

```
$credentials = [Amazon.Runtime.InstanceProfileAWSCredentials]::new()
Get-SSMDeployablePatchSnapshotForInstance -SnapshotId "4681775b-098f-4435-
a956-0ef33373ac11" -InstanceId "i-0cb2b964d3e14fd9f" -Credentials $credentials
```

Sortie :

```
InstanceId          SnapshotDownloadUrl
----------          -------------------
i-0cb2b964d3e14fd9f https://patch-baseline-snapshot-us-west-2.s3-us-
west-2.amazonaws.com/853d0d3db0f0cafe...1692/4681775b-098f-4435...
```

Exemple 2 : Cet exemple montre comment obtenir le résultat complet SnapshotDownloadUrl. Cette commande doit être exécutée depuis l'instance à l'aide des informations d'identification de l'instance. Pour s'assurer qu'elle utilise les informations d'identification de l'instance, l'exemple configure la PowerShell session pour qu'elle utilise un **`Amazon.Runtime.InstanceProfileAWSCredentials`** objet.

```
Set-AWSCredential -Credential
 ([Amazon.Runtime.InstanceProfileAWSCredentials]::new())
(Get-SSMDeployablePatchSnapshotForInstance -SnapshotId "4681775b-098f-4435-
a956-0ef33373ac11" -InstanceId "i-0cb2b964d3e14fd9f").SnapshotDownloadUrl
```

Sortie :

```
https://patch-baseline-snapshot-us-west-2.s3-us-
west-2.amazonaws.com/853d0d3db0f0cafe...
```

- Pour plus de détails sur l'API, reportez-vous GetDeployablePatchSnapshotForInstanceà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`GetDocument`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`GetDocument`.

CLI

AWS CLI

Pour obtenir le contenu d'un document

L'`get-document`exemple suivant affiche le contenu d'un document Systems Manager.

```
aws ssm get-document \
    --name "AWS-RunShellScript"
```

Sortie :

```
{
    "Name": "AWS-RunShellScript",
    "DocumentVersion": "1",
    "Status": "Active",
    "Content": "{\n    \"schemaVersion\":\"1.2\",\n    \"description\":\"Run
 a shell script or specify the commands to run.\",\n    \"parameters\":{\n
        \"commands\":{\n            \"type\":\"StringList\",\n
 \"description\":\"(Required) Specify a shell script or a command to run.\",
\n            \"minItems\":1,\n            \"displayType\":\"textarea\"\n
      },\n        \"workingDirectory\":{\n            \"type\":\"String\",\n
            \"default\":\"\",\n            \"description\":\"(Optional) The
 path to the working directory on your instance.\",\n            \"maxChars
\":4096\n        },\n        \"executionTimeout\":{\n            \"type\":
\"String\",\n            \"default\":\"3600\",\n            \"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).\",\n            \"allowedPattern\":\"([1-9][0-9]{0,4})|(1[0-6][0-9]
{4})|(17[0-1][0-9]{3})|(172[0-7][0-9]{2})|(172800)\"\n        }\n    },\n
 \"runtimeConfig\":{\n        \"aws:runShellScript\":{\n            \"properties
\":[\n                {\n                    \"id\":\"0.aws:runShellScript
\",\n                    \"runCommand\":\"{{ commands }}\",\n
```

```
      \"workingDirectory\":\"{{ workingDirectory }}\",\n
 \"timeoutSeconds\":\"{{ executionTimeout }}\"\n                }\n
  ]\n        }\n    }\n}\n",
    "DocumentType": "Command",
    "DocumentFormat": "JSON"
}
```

Pour plus d'informations, consultez les documents de AWS Systems Manager dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous GetDocumentà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple renvoie le contenu d'un document.

```
Get-SSMDocument -Name "RunShellScript"
```

Sortie :

```
Content
-------
{...
```

Exemple 2 : Cet exemple affiche le contenu complet d'un document.

```
(Get-SSMDocument -Name "RunShellScript").Content
{
   "schemaVersion":"2.0",
   "description":"Run an updated script",
   "parameters":{
      "commands":{
         "type":"StringList",
         "description":"(Required) Specify a shell script or a command to run.",
         "minItems":1,
         "displayType":"textarea"
      }
   },
```

```
   "mainSteps":[
      {
         "action":"aws:runShellScript",
         "name":"runShellScript",
         "inputs":{
            "commands":"{{ commands }}"
         }
      },
      {
         "action":"aws:runPowerShellScript",
         "name":"runPowerShellScript",
         "inputs":{
            "commands":"{{ commands }}"
         }
      }
   ]
}
```

- Pour plus de détails sur l'API, reportez-vous GetDocumentà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`GetInventory`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`GetInventory`.

CLI

AWS CLI

Pour consulter votre inventaire

Cet exemple permet d'obtenir les métadonnées personnalisées de votre inventaire.

Commande :

```
aws ssm get-inventory
```

Sortie :

```
{
  "Entities": [
      {
          "Data": {
              "AWS:InstanceInformation": {
                  "Content": [
                      {
                          "ComputerName": "ip-172-31-44-222.us-
west-2.compute.internal",
                          "InstanceId": "i-0cb2b964d3e14fd9f",
                          "IpAddress": "172.31.44.222",
                          "AgentType": "amazon-ssm-agent",
                          "ResourceType": "EC2Instance",
                          "AgentVersion": "2.0.672.0",
                          "PlatformVersion": "2016.09",
                          "PlatformName": "Amazon Linux AMI",
                          "PlatformType": "Linux"
                      }
                  ],
                  "TypeName": "AWS:InstanceInformation",
                  "SchemaVersion": "1.0",
                  "CaptureTime": "2017-02-20T18:03:58Z"
              }
          },
          "Id": "i-0cb2b964d3e14fd9f"
      }
  ]
}
```

- Pour plus de détails sur l'API, reportez-vous GetInventoryà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple permet d'obtenir les métadonnées personnalisées de votre inventaire.

```
Get-SSMInventory
```

Sortie :

```
Data
     Id
----
     --
{[AWS:InstanceInformation,
 Amazon.SimpleSystemsManagement.Model.InventoryResultItem]} i-0cb2b964d3e14fd9f
```

- Pour plus de détails sur l'API, reportez-vous GetInventoryà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`GetInventorySchema`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`GetInventorySchema`.

CLI

AWS CLI

Pour consulter votre schéma d'inventaire

Cet exemple renvoie une liste de noms de types d'inventaire pour le compte.

Commande :

```
aws ssm get-inventory-schema
```

Sortie :

```
{
  "Schemas": [
      {
          "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"
            }
        ]
    },
    ...
  ],
  "NextToken": "--token string truncated--"
}
```

Pour consulter le schéma d'inventaire pour un type d'inventaire spécifique

Cet exemple renvoie le schéma d'inventaire pour un type d'inventaire des AWS composants AWS :.

Commande :

```
aws ssm get-inventory-schema --type-name "AWS:AWSComponent"
```

- Pour plus de détails sur l'API, voir GetInventorySchéma dans AWS CLI la référence des commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple renvoie une liste de noms de types d'inventaire pour le compte.

```
Get-SSMInventorySchema
```

- Pour plus de détails sur l'API, consultez la section GetInventorySchéma dans la référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`GetMaintenanceWindow`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`GetMaintenanceWindow`.

CLI

AWS CLI

Pour obtenir des informations sur une fenêtre de maintenance

L'`get-maintenance-window`exemple suivant permet de récupérer les détails relatifs à la fenêtre de maintenance spécifiée.

```
aws ssm get-maintenance-window \
    --window-id "mw-03eb9db428EXAMPLE"
```

Sortie :

```
{
    "AllowUnassociatedTargets": true,
    "CreatedDate": 1515006912.957,
    "Cutoff": 1,
    "Duration": 6,
    "Enabled": true,
    "ModifiedDate": 2020-01-01T10:04:04.099Z,
```

```
    "Name": "My-Maintenance-Window",
    "Schedule": "rate(3 days)",
    "WindowId": "mw-03eb9db428EXAMPLE",
    "NextExecutionTime": "2020-02-25T00:08:15.099Z"
}
```

Pour plus d'informations, consultez la section Afficher les informations sur les fenêtres de maintenance (AWS CLI) dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, voir GetMaintenanceWindow in AWS CLI Command Reference.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple permet d'obtenir des informations sur une fenêtre de maintenance.

```
Get-SSMMaintenanceWindow -WindowId "mw-03eb9db42890fb82d"
```

Sortie :

```
AllowUnassociatedTargets : False
CreatedDate              : 2/20/2017 6:14:05 PM
Cutoff                   : 1
Duration                 : 2
Enabled                  : True
ModifiedDate             : 2/20/2017 6:14:05 PM
Name                     : TestMaintWin
Schedule                 : cron(0 */30 * * * ? *)
WindowId                 : mw-03eb9db42890fb82d
```

- Pour plus de détails sur l'API, consultez GetMaintenancela section Window in AWS Tools for PowerShell Cmdlet Reference.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **GetMaintenanceWindowExecution** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`GetMaintenanceWindowExecution`.

CLI

AWS CLI

Pour obtenir des informations sur l'exécution d'une tâche dans une fenêtre de maintenance

L'`get-maintenance-window-execution`exemple suivant répertorie les informations relatives à une tâche exécutée dans le cadre de l'exécution de la fenêtre de maintenance spécifiée.

```
aws ssm get-maintenance-window-execution \
    --window-execution-id "518d5565-5969-4cca-8f0e-da3b2EXAMPLE"
```

Sortie :

```
{
    "Status": "SUCCESS",
    "TaskIds": [
        "ac0c6ae1-daa3-4a89-832e-d3845EXAMPLE"
    ],
    "StartTime": 1487692834.595,
    "EndTime": 1487692835.051,
    "WindowExecutionId": "518d5565-5969-4cca-8f0e-da3b2EXAMPLE",
}
```

Pour plus d'informations, consultez la section Afficher les informations sur les tâches et les exécutions de tâches (AWS CLI) dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous GetMaintenanceWindowExecutionà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple répertorie les informations relatives à une tâche exécutée dans le cadre de l'exécution d'une fenêtre de maintenance.

```
Get-SSMMaintenanceWindowExecution -WindowExecutionId "518d5565-5969-4cca-8f0e-
da3b2a638355"
```

Sortie :

```
EndTime           : 2/21/2017 4:00:35 PM
StartTime         : 2/21/2017 4:00:34 PM
Status            : FAILED
StatusDetails     : One or more tasks in the orchestration failed.
TaskIds           : {ac0c6ae1-daa3-4a89-832e-d384503b6586}
WindowExecutionId : 518d5565-5969-4cca-8f0e-da3b2a638355
```

- Pour plus de détails sur l'API, reportez-vous GetMaintenanceWindowExecutionà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`GetMaintenanceWindowExecutionTask`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`GetMaintenanceWindowExecutionTask`.

CLI

AWS CLI

Pour obtenir des informations sur l'exécution d'une tâche dans une fenêtre de maintenance

L'`get-maintenance-window-execution-task`exemple suivant répertorie les informations relatives à une tâche faisant partie de l'exécution de la fenêtre de maintenance spécifiée.

```
aws ssm get-maintenance-window-execution-task \
    --window-execution-id "518d5565-5969-4cca-8f0e-da3b2EXAMPLE" \
    --task-id "ac0c6ae1-daa3-4a89-832e-d3845EXAMPLE"
```

Sortie :

```
{
    "WindowExecutionId": "518d5565-5969-4cca-8f0e-da3b2EXAMPLE",
    "TaskExecutionId": "ac0c6ae1-daa3-4a89-832e-d3845EXAMPLE",
    "TaskArn": "AWS-RunPatchBaseline",
    "ServiceRole": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
    "Type": "RUN_COMMAND",
    "TaskParameters": [
        {
            "BaselineOverride": {
                "Values": [
                    ""
                ]
            },
            "InstallOverrideList": {
                "Values": [
                    ""
                ]
            },
            "Operation": {
                "Values": [
                    "Scan"
                ]
            },
            "RebootOption": {
                "Values": [
                    "RebootIfNeeded"
                ]
            },
            "SnapshotId": {
                "Values": [
                    "{{ aws:ORCHESTRATION_ID }}"
                ]
            },
            "aws:InstanceId": {
                "Values": [
                    "i-02573cafcfEXAMPLE",
```

```
                    "i-0471e04240EXAMPLE",
                    "i-07782c72faEXAMPLE"
                ]
            }
        }
    ],
    "Priority": 1,
    "MaxConcurrency": "1",
    "MaxErrors": "3",
    "Status": "SUCCESS",
    "StartTime": "2021-08-04T11:45:35.088000-07:00",
    "EndTime": "2021-08-04T11:53:09.079000-07:00"
}
```

Pour plus d'informations, consultez la section Afficher les informations relatives aux tâches et aux exécutions de tâches (AWS CLI) dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, consultez la section GetMaintenanceWindowExecutionTask in AWS CLI Command Reference.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple répertorie les informations relatives à une tâche qui faisait partie de l'exécution d'une fenêtre de maintenance.

```
Get-SSMMaintenanceWindowExecutionTask -TaskId "ac0c6ae1-daa3-4a89-832e-
d384503b6586" -WindowExecutionId "518d5565-5969-4cca-8f0e-da3b2a638355"
```

Sortie :

```
EndTime           : 2/21/2017 4:00:35 PM
MaxConcurrency    : 1
MaxErrors         : 1
Priority          : 10
ServiceRole       : arn:aws:iam::123456789012:role/MaintenanceWindowsRole
StartTime         : 2/21/2017 4:00:34 PM
Status            : FAILED
StatusDetails     : The maximum error count was exceeded.
TaskArn           : AWS-RunShellScript
TaskExecutionId   : ac0c6ae1-daa3-4a89-832e-d384503b6586
```

```
TaskParameters    :
 {Amazon.Runtime.Internal.Util.AlwaysSendDictionary`2[System.String,Amazon.SimpleSystemsM
                    meterValueExpression]}
Type              : RUN_COMMAND
WindowExecutionId : 518d5565-5969-4cca-8f0e-da3b2a638355
```

- Pour plus de détails sur l'API, consultez la section GetMaintenanceWindowExecutionTask in AWS Tools for PowerShell Cmdlet Reference.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`GetParameterHistory`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`GetParameterHistory`.

CLI

AWS CLI

Pour obtenir l'historique des valeurs d'un paramètre

L'`get-parameter-history`exemple suivant répertorie l'historique des modifications apportées au paramètre spécifié, y compris sa valeur.

```
aws ssm get-parameter-history \
    --name "MyStringParameter"
```

Sortie :

```
{
    "Parameters": [
        {
            "Name": "MyStringParameter",
            "Type": "String",
            "LastModifiedDate": 1582154711.976,
            "LastModifiedUser": "arn:aws:iam::111222333444:user/Mary-Major",
            "Description": "This is the first version of my String parameter",
            "Value": "Veni",
```

```
            "Version": 1,
            "Labels": [],
            "Tier": "Standard",
            "Policies": []
        },
        {
            "Name": "MyStringParameter",
            "Type": "String",
            "LastModifiedDate": 1582156093.471,
            "LastModifiedUser": "arn:aws:iam::111222333444:user/Mary-Major",
            "Description": "This is the second version of my String parameter",
            "Value": "Vidi",
            "Version": 2,
            "Labels": [],
            "Tier": "Standard",
            "Policies": []
        },
        {
            "Name": "MyStringParameter",
            "Type": "String",
            "LastModifiedDate": 1582156117.545,
            "LastModifiedUser": "arn:aws:iam::111222333444:user/Mary-Major",
            "Description": "This is the third version of my String parameter",
            "Value": "Vici",
            "Version": 3,
            "Labels": [],
            "Tier": "Standard",
            "Policies": []
        }
    ]
}
```

Pour plus d'informations, consultez la section Travailler avec des versions de paramètres dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, consultez la section GetParameterHistorique dans AWS CLI Command Reference.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple répertorie l'historique des valeurs d'un paramètre.

```
Get-SSMParameterHistory -Name "Welcome"
```

Sortie :

```
Description      :
KeyId            :
LastModifiedDate : 3/3/2017 6:55:25 PM
LastModifiedUser : arn:aws:iam::123456789012:user/admin
Name             : Welcome
Type             : String
Value            : helloWorld
```

- Pour plus de détails sur l'API, consultez la section GetParameterHistorique dans la référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`GetParameters`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`GetParameters`.

CLI

AWS CLI

Exemple 1 : pour répertorier les valeurs d'un paramètre

L'`get-parameters`exemple suivant répertorie les valeurs des trois paramètres spécifiés.

```
aws ssm get-parameters \
    --names "MyStringParameter" "MyStringListParameter" "MyInvalidParameterName"
```

Sortie :

```
{
    "Parameters": [
        {
            "Name": "MyStringListParameter",
```

```
            "Type": "StringList",
            "Value": "alpha,beta,gamma",
            "Version": 1,
            "LastModifiedDate": 1582154764.222,
            "ARN": "arn:aws:ssm:us-east-2:111222333444:parameter/
MyStringListParameter"
            "DataType": "text"
        },
        {
            "Name": "MyStringParameter",
            "Type": "String",
            "Value": "Vici",
            "Version": 3,
            "LastModifiedDate": 1582156117.545,
            "ARN": "arn:aws:ssm:us-east-2:111222333444:parameter/
MyStringParameter"
            "DataType": "text"
        }
    ],
    "InvalidParameters": [
        "MyInvalidParameterName"
    ]
}
```

Pour plus d'informations, reportez-vous à la section Working with Parameter Store dans le guide de l'utilisateur de AWS Systems Manager.

Exemple 2 : Pour répertorier les noms et les valeurs de plusieurs paramètres à l'aide de l'option ``--query``

L'`get-parameters`exemple suivant répertorie les noms et les valeurs des paramètres spécifiés.

```
aws ssm get-parameters \
    --names MyStringParameter MyStringListParameter \
    --query "Parameters[*].{Name:Name,Value:Value}"
```

Sortie :

```
[
    {
        "Name": "MyStringListParameter",
```

```
        "Value": "alpha,beta,gamma"
    },
    {
        "Name": "MyStringParameter",
        "Value": "Vidi"
    }
]
```

Pour plus d'informations, reportez-vous à la section Working with Parameter Store dans le guide de l'utilisateur de AWS Systems Manager.

Exemple 3 : pour afficher la valeur d'un paramètre à l'aide d'étiquettes

L'`get-parameter`exemple suivant répertorie la valeur du paramètre unique spécifié avec une étiquette spécifiée.

```
aws ssm get-parameter \
    --name "MyParameter:label"
```

Sortie :

```
{
    "Parameters": [
        {
            "Name": "MyLabelParameter",
            "Type": "String",
            "Value": "parameter by label",
            "Version": 1,
            "Selector": ":label",
            "LastModifiedDate": "2021-07-12T09:49:15.865000-07:00",
            "ARN": "arn:aws:ssm:us-west-2:786973925828:parameter/MyParameter",
            "DataType": "text"
        },
        {
            "Name": "MyVersionParameter",
            "Type": "String",
            "Value": "parameter by version",
            "Version": 2,
            "Selector": ":2",
            "LastModifiedDate": "2021-03-24T16:20:28.236000-07:00",
            "ARN": "arn:aws:ssm:us-west-2:786973925828:parameter/unlabel-param",
            "DataType": "text"
        }
```

```
    ],
    "InvalidParameters": []
}
```

Pour plus d'informations, reportez-vous à la section Utilisation des libellés de paramètres dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous GetParametersà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple répertorie les valeurs d'un paramètre.

```
Get-SSMParameterValue -Name "Welcome"
```

Sortie :

```
InvalidParameters Parameters
----------------- ----------
{}                {Welcome}
```

Exemple 2 : Cet exemple répertorie les détails de la valeur.

```
(Get-SSMParameterValue -Name "Welcome").Parameters
```

Sortie :

```
Name    Type   Value
----    ----   -----
Welcome String Good day, Sunshine!
```

- Pour plus de détails sur l'API, reportez-vous GetParametersà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **GetPatchBaseline** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`GetPatchBaseline`.

CLI

AWS CLI

Pour afficher une ligne de base de correctif

L'`get-patch-baseline`exemple suivant récupère les détails de la ligne de base de correctif spécifiée.

```
aws ssm get-patch-baseline \
    --baseline-id "pb-0123456789abcdef0"
```

Sortie :

```
{
    "BaselineId": "pb-0123456789abcdef0",
    "Name": "WindowsPatching",
    "OperatingSystem": "WINDOWS",
    "GlobalFilters": {
        "PatchFilters": []
    },
    "ApprovalRules": {
        "PatchRules": [
            {
                "PatchFilterGroup": {
                    "PatchFilters": [
                        {
                            "Key": "PRODUCT",
                            "Values": [
                                "WindowsServer2016"
                            ]
                        }
                    ]
                },
                "ComplianceLevel": "CRITICAL",
                "ApproveAfterDays": 0,
                "EnableNonSecurity": false
            }
```

```
        ]
    },
    "ApprovedPatches": [],
    "ApprovedPatchesComplianceLevel": "UNSPECIFIED",
    "ApprovedPatchesEnableNonSecurity": false,
    "RejectedPatches": [],
    "RejectedPatchesAction": "ALLOW_AS_DEPENDENCY",
    "PatchGroups": [
        "QA",
        "DEV"
    ],
    "CreatedDate": 1550244180.465,
    "ModifiedDate": 1550244180.465,
    "Description": "Patches for Windows Servers",
    "Sources": []
}
```

Pour plus d'informations, consultez la section À propos des lignes de base de correctifs dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, voir GetPatchBaseline dans AWS CLI Command Reference.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple affiche les détails d'une ligne de base de correctifs.

```
Get-SSMPatchBaselineDetail -BaselineId "pb-03da896ca3b68b639"
```

Sortie :

```
ApprovalRules   : Amazon.SimpleSystemsManagement.Model.PatchRuleGroup
ApprovedPatches : {}
BaselineId      : pb-03da896ca3b68b639
CreatedDate     : 3/3/2017 5:02:19 PM
Description     : Baseline containing all updates approved for production systems
GlobalFilters   : Amazon.SimpleSystemsManagement.Model.PatchFilterGroup
ModifiedDate    : 3/3/2017 5:02:19 PM
Name            : Production-Baseline
PatchGroups     : {}
RejectedPatches : {}
```

- Pour plus de détails sur l'API, consultez la section GetPatchBaseline dans la référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **GetPatchBaselineForPatchGroup** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`GetPatchBaselineForPatchGroup`.

CLI

AWS CLI

Pour afficher la ligne de base de correctifs pour un groupe de correctifs

L'`get-patch-baseline-for-patch-group`exemple suivant permet de récupérer des informations sur la ligne de base de correctifs pour le groupe de correctifs spécifié.

```
aws ssm get-patch-baseline-for-patch-group \
    --patch-group "DEV"
```

Sortie :

```
{
    "PatchGroup": "DEV",
    "BaselineId": "pb-0123456789abcdef0",
    "OperatingSystem": "WINDOWS"
}
```

Pour plus d'informations, consultez les sections Create a Patch Group__ < https://docs.aws.amazon.com/systems-manager/latest/userguide/sysman-patch-group-tagging.html> et Add a Patch Group to a Patch Baseline dans le Guide de l'utilisateur de Systems AWS Manager.

- Pour plus de détails sur l'API, reportez-vous GetPatchBaselineForPatchGroupà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : cet exemple affiche la ligne de base de correctifs pour un groupe de correctifs.

```
Get-SSMPatchBaselineForPatchGroup -PatchGroup "Production"
```

Sortie :

```
BaselineId           PatchGroup
----------           ----------
pb-045f10b4f382baeda Production
```

- Pour plus de détails sur l'API, reportez-vous GetPatchBaselineForPatchGroupà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`ListAssociationVersions`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`ListAssociationVersions`.

CLI

AWS CLI

Pour répertorier toutes les versions d'une association pour un ID d'association spécifique

L'`list-association-versions`exemple suivant répertorie toutes les versions des associations spécifiées.

```
aws ssm list-association-versions \
    --association-id "8dfe3659-4309-493a-8755-0123456789ab"
```

Sortie :

```
{
"AssociationVersions": [
```

```
        {
            "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
            "AssociationVersion": "1",
            "CreatedDate": 1550505536.726,
            "Name": "AWS-UpdateSSMAgent",
            "Parameters": {
                "allowDowngrade": [
                    "false"
                ],
                "version": [
                    ""
                ]
            },
            "Targets": [
                {
                    "Key": "InstanceIds",
                    "Values": [
                        "i-1234567890abcdef0"
                    ]
                }
            ],
            "ScheduleExpression": "cron(0 00 12 ? * SUN *)",
            "AssociationName": "UpdateSSMAgent"
        }
    ]
}
```

Pour plus d'informations, reportez-vous à la section Utilisation des associations dans Systems Manager dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, voir ListAssociationVersions dans AWS CLI Command Reference.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple récupère toutes les versions de l'association fournie.

```
Get-SSMAssociationVersionList -AssociationId 123a45a0-c678-9012-3456-78901234db5e
```

Sortie :

```
AssociationId      : 123a45a0-c678-9012-3456-78901234db5e
AssociationName    :
AssociationVersion : 2
ComplianceSeverity :
CreatedDate        : 3/12/2019 9:21:01 AM
DocumentVersion    :
MaxConcurrency     :
MaxErrors          :
Name               : AWS-GatherSoftwareInventory
OutputLocation     :
Parameters         : {}
ScheduleExpression :
Targets            : {InstanceIds}

AssociationId      : 123a45a0-c678-9012-3456-78901234db5e
AssociationName    : test-case-1234567890
AssociationVersion : 1
ComplianceSeverity :
CreatedDate        : 3/2/2019 8:53:29 AM
DocumentVersion    :
MaxConcurrency     :
MaxErrors          :
Name               : AWS-GatherSoftwareInventory
OutputLocation     :
Parameters         : {}
ScheduleExpression : rate(30minutes)
Targets            : {InstanceIds}
```

- Pour plus de détails sur l'API, consultez la section ListAssociationVersions dans la référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`ListAssociations`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`ListAssociations`.

CLI

AWS CLI

Exemple 1 : pour répertorier vos associations pour une instance spécifique

L'exemple de list-associations suivant répertorie toutes les associations avec AssociationName UpdatessMagent.

```
aws ssm list-associations /
    --association-filter-list "key=AssociationName,value=UpdateSSMAgent"
```

Sortie :

```
{
    "Associations": [
        {
            "Name": "AWS-UpdateSSMAgent",
            "InstanceId": "i-1234567890abcdef0",
            "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
            "AssociationVersion": "1",
            "Targets": [
                {
                    "Key": "InstanceIds",
                    "Values": [
                        "i-016648b75dd622dab"
                    ]
                }
            ],
            "Overview": {
                "Status": "Pending",
                "DetailedStatus": "Associated",
                "AssociationStatusAggregatedCount": {
                    "Pending": 1
                }
            },
            "ScheduleExpression": "cron(0 00 12 ? * SUN *)",
            "AssociationName": "UpdateSSMAgent"
        }
    ]
}
```

Pour plus d'informations, reportez-vous à la section Utilisation des associations dans Systems Manager dans le Guide de l'utilisateur de Systems Manager.

Exemple 2 : pour répertorier vos associations pour un document spécifique

L'exemple de liste-associations suivant répertorie toutes les associations pour le document spécifié.

```
aws ssm list-associations /
    --association-filter-list "key=Name,value=AWS-UpdateSSMAgent"
```

Sortie :

```
{
    "Associations": [
        {
            "Name": "AWS-UpdateSSMAgent",
            "InstanceId": "i-1234567890abcdef0",
            "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
            "AssociationVersion": "1",
            "Targets": [
                {
                    "Key": "InstanceIds",
                    "Values": [
                        "i-1234567890abcdef0"
                    ]
                }
            ],
            "LastExecutionDate": 1550505828.548,
            "Overview": {
                "Status": "Success",
                "DetailedStatus": "Success",
                "AssociationStatusAggregatedCount": {
                    "Success": 1
                }
            },
            "ScheduleExpression": "cron(0 00 12 ? * SUN *)",
            "AssociationName": "UpdateSSMAgent"
        },
    {
            "Name": "AWS-UpdateSSMAgent",
            "InstanceId": "i-9876543210abcdef0",
            "AssociationId": "fbc07ef7-b985-4684-b82b-0123456789ab",
```

```
            "AssociationVersion": "1",
            "Targets": [
                {
                    "Key": "InstanceIds",
                    "Values": [
                        "i-9876543210abcdef0"
                    ]
                }
            ],
            "LastExecutionDate": 1550507531.0,
            "Overview": {
                "Status": "Success",
                "AssociationStatusAggregatedCount": {
                    "Success": 1
                }
            }
        }
    ]
}
```

Pour plus d'informations, reportez-vous à la section Utilisation des associations dans Systems Manager dans le Guide de l'utilisateur de Systems Manager.

- Pour plus de détails sur l'API, reportez-vous ListAssociationsà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple répertorie toutes les associations associées à une instance. La syntaxe utilisée dans cet exemple nécessite PowerShell la version 3 ou ultérieure.

```
$filter1 = @{Key="InstanceId";Value=@("i-0000293ffd8c57862")}
Get-SSMAssociationList -AssociationFilterList $filter1
```

Sortie :

```
AssociationId      : d8617c07-2079-4c18-9847-1655fc2698b0
DocumentVersion    :
InstanceId         : i-0000293ffd8c57862
LastExecutionDate  : 2/20/2015 8:31:11 AM
```

```
Name               : AWS-UpdateSSMAgent
Overview           : Amazon.SimpleSystemsManagement.Model.AssociationOverview
ScheduleExpression :
Targets            : {InstanceIds}
```

Exemple 2 : Cet exemple répertorie toutes les associations associées à un document de configuration. La syntaxe utilisée dans cet exemple nécessite PowerShell la version 3 ou ultérieure.

```
$filter2 = @{Key="Name";Value=@("AWS-UpdateSSMAgent")}
Get-SSMAssociationList -AssociationFilterList $filter2
```

Sortie :

```
AssociationId      : d8617c07-2079-4c18-9847-1655fc2698b0
DocumentVersion    :
InstanceId         : i-0000293ffd8c57862
LastExecutionDate  : 2/20/2015 8:31:11 AM
Name               : AWS-UpdateSSMAgent
Overview           : Amazon.SimpleSystemsManagement.Model.AssociationOverview
ScheduleExpression :
Targets            : {InstanceIds}
```

Exemple 3 : Avec PowerShell la version 2, vous devez utiliser New-Object pour créer chaque filtre.

```
$filter1 = New-Object Amazon.SimpleSystemsManagement.Model.AssociationFilter
$filter1.Key = "InstanceId"
$filter1.Value = "i-0000293ffd8c57862"

Get-SSMAssociationList -AssociationFilterList $filter1
```

Sortie :

```
AssociationId      : d8617c07-2079-4c18-9847-1655fc2698b0
DocumentVersion    :
InstanceId         : i-0000293ffd8c57862
LastExecutionDate  : 2/20/2015 8:31:11 AM
Name               : AWS-UpdateSSMAgent
Overview           : Amazon.SimpleSystemsManagement.Model.AssociationOverview
ScheduleExpression :
```

```
Targets            : {InstanceIds}
```

- Pour plus de détails sur l'API, reportez-vous ListAssociationsà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`ListCommandInvocations`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`ListCommandInvocations`.

CLI

AWS CLI

Pour répertorier les appels d'une commande spécifique

L'`list-command-invocations`exemple suivant répertorie toutes les invocations d'une commande.

```
aws ssm list-command-invocations \
    --command-id "ef7fdfd8-9b57-4151-a15c-db9a12345678" \
    --details
```

Sortie :

```
{
    "CommandInvocations": [
        {
            "CommandId": "ef7fdfd8-9b57-4151-a15c-db9a12345678",
            "InstanceId": "i-02573cafcfEXAMPLE",
            "InstanceName": "",
            "Comment": "b48291dd-ba76-43e0-
b9df-13e11ddaac26:6960febb-2907-4b59-8e1a-d6ce8EXAMPLE",
            "DocumentName": "AWS-UpdateSSMAgent",
            "DocumentVersion": "",
            "RequestedDateTime": 1582136283.089,
            "Status": "Success",
            "StatusDetails": "Success",
            "StandardOutputUrl": "",
```

```
            "StandardErrorUrl": "",
            "CommandPlugins": [
                {
                    "Name": "aws:updateSsmAgent",
                    "Status": "Success",
                    "StatusDetails": "Success",
                    "ResponseCode": 0,
                    "ResponseStartDateTime": 1582136283.419,
                    "ResponseFinishDateTime": 1582136283.51,
                    "Output": "Updating amazon-ssm-agent from 2.3.842.0 to latest
\nSuccessfully downloaded https://s3.us-east-2.amazonaws.com/amazon-ssm-us-
east-2/ssm-agent-manifest.json\namazon-ssm-agent 2.3.842.0 has already been
 installed, update skipped\n",
                    "StandardOutputUrl": "",
                    "StandardErrorUrl": "",
                    "OutputS3Region": "us-east-2",
                    "OutputS3BucketName": "",
                    "OutputS3KeyPrefix": ""
                }
            ],
            "ServiceRole": "",
            "NotificationConfig": {
                "NotificationArn": "",
                "NotificationEvents": [],
                "NotificationType": ""
            },
            "CloudWatchOutputConfig": {
                "CloudWatchLogGroupName": "",
                "CloudWatchOutputEnabled": false
            }
        },
        {
            "CommandId": "ef7fdfd8-9b57-4151-a15c-db9a12345678",
            "InstanceId": "i-0471e04240EXAMPLE",
            "InstanceName": "",
            "Comment": "b48291dd-ba76-43e0-
b9df-13e11ddaac26:6960febb-2907-4b59-8e1a-d6ce8EXAMPLE",
            "DocumentName": "AWS-UpdateSSMAgent",
            "DocumentVersion": "",
            "RequestedDateTime": 1582136283.02,
            "Status": "Success",
            "StatusDetails": "Success",
            "StandardOutputUrl": "",
            "StandardErrorUrl": "",
```

```
            "CommandPlugins": [
                {
                    "Name": "aws:updateSsmAgent",
                    "Status": "Success",
                    "StatusDetails": "Success",
                    "ResponseCode": 0,
                    "ResponseStartDateTime": 1582136283.812,
                    "ResponseFinishDateTime": 1582136295.031,
                    "Output": "Updating amazon-ssm-agent from 2.3.672.0 to
 latest\nSuccessfully downloaded https://s3.us-east-2.amazonaws.com/amazon-
ssm-us-east-2/ssm-agent-manifest.json\nSuccessfully downloaded https://s3.us-
east-2.amazonaws.com/amazon-ssm-us-east-2/amazon-ssm-agent-updater/2.3.842.0/
amazon-ssm-agent-updater-snap-amd64.tar.gz\nSuccessfully downloaded https://
s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/amazon-ssm-agent/2.3.672.0/
amazon-ssm-agent-snap-amd64.tar.gz\nSuccessfully downloaded https://s3.us-
east-2.amazonaws.com/amazon-ssm-us-east-2/amazon-ssm-agent/2.3.842.0/amazon-ssm-
agent-snap-amd64.tar.gz\nInitiating amazon-ssm-agent update to 2.3.842.0\namazon-
ssm-agent updated successfully to 2.3.842.0",
                    "StandardOutputUrl": "",
                    "StandardErrorUrl": "",
                    "OutputS3Region": "us-east-2",
                    "OutputS3BucketName": "",
                    "OutputS3KeyPrefix": "8bee3135-398c-4d31-99b6-e42d2EXAMPLE/
i-0471e04240EXAMPLE/awsupdateSsmAgent"
                }
            ],
            "ServiceRole": "",
            "NotificationConfig": {
                "NotificationArn": "",
                "NotificationEvents": [],
                "NotificationType": ""
            },
            "CloudWatchOutputConfig": {
                "CloudWatchLogGroupName": "",
                "CloudWatchOutputEnabled": false
            }
        }
    ]
}
```


Pour plus d'informations, consultez la section Understanding Command Statuses dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, consultez la section ListCommandInvocations dans la référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple répertorie toutes les invocations d'une commande.

```
Get-SSMCommandInvocation -CommandId "b8eac879-0541-439d-94ec-47a80d554f44" -
Detail $true
```

Sortie :

```
CommandId          : b8eac879-0541-439d-94ec-47a80d554f44
CommandPlugins     : {aws:runShellScript}
Comment            : IP config
DocumentName       : AWS-RunShellScript
InstanceId         : i-0cb2b964d3e14fd9f
InstanceName       :
NotificationConfig : Amazon.SimpleSystemsManagement.Model.NotificationConfig
RequestedDateTime  : 2/22/2017 8:13:16 PM
ServiceRole        :
StandardErrorUrl   :
StandardOutputUrl  :
Status             : Success
StatusDetails      : Success
TraceOutput        :
```

Exemple 2 : Cet exemple répertorie CommandPlugins pour l'invocation de l'identifiant de commande e1eb2e3c-ed4c-5123-45c1-234f5612345f

```
Get-SSMCommandInvocation -CommandId e1eb2e3c-ed4c-5123-45c1-234f5612345f -Detail:
$true | Select-Object -ExpandProperty CommandPlugins
```

Sortie :

```
Name                  : aws:runPowerShellScript
Output                : Completed 17.7 KiB/17.7 KiB (40.1 KiB/s) with 1 file(s)
 remainingdownload: s3://dd-aess-r-ctmer/KUMO.png to ..\..\programdata\KUMO.png
                        kumo available
```

```
OutputS3BucketName     :
OutputS3KeyPrefix      :
OutputS3Region         : eu-west-1
ResponseCode           : 0
ResponseFinishDateTime : 4/3/2019 11:53:23 AM
ResponseStartDateTime  : 4/3/2019 11:53:21 AM
StandardErrorUrl       :
StandardOutputUrl      :
Status                 : Success
StatusDetails          : Success
```

- Pour plus de détails sur l'API, consultez la section ListCommandInvocations dans la référence des applets de AWS Tools for PowerShell commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`ListCommands`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`ListCommands`.

CLI

AWS CLI

Exemple 1 : pour obtenir le statut d'une commande spécifique

L'`list-commands`exemple suivant extrait et affiche le statut de la commande spécifiée.

```
aws ssm list-commands \
    --command-id "0831e1a8-a1ac-4257-a1fd-c831bEXAMPLE"
```

Exemple 2 : Pour obtenir le statut des commandes demandées après une date précise

L'`list-commands`exemple suivant récupère les détails des commandes demandées après la date spécifiée.

```
aws ssm list-commands \
    --filter "key=InvokedAfter,value=2020-02-01T00:00:00Z"
```

Exemple 3 : Pour répertorier toutes les commandes demandées dans un AWS compte

L'`list-commands`exemple suivant répertorie toutes les commandes demandées par les utilisateurs du AWS compte courant et de la région.

```
aws ssm list-commands
```

Sortie :

```
{
    "Commands": [
        {
            "CommandId": "8bee3135-398c-4d31-99b6-e42d2EXAMPLE",
            "DocumentName": "AWS-UpdateSSMAgent",
            "DocumentVersion": "",
            "Comment": "b48291dd-ba76-43e0-
b9df-13e11ddaac26:6960febb-2907-4b59-8e1a-d6ce8EXAMPLE",
            "ExpiresAfter": "2020-02-19T11:28:02.500000-08:00",
            "Parameters": {},
            "InstanceIds": [
                "i-028ea792daEXAMPLE",
                "i-02feef8c46EXAMPLE",
                "i-038613f3f0EXAMPLE",
                "i-03a530a2d4EXAMPLE",
                "i-083b678d37EXAMPLE",
                "i-0dee81debaEXAMPLE"
            ],
            "Targets": [],
            "RequestedDateTime": "2020-02-19T10:18:02.500000-08:00",
            "Status": "Success",
            "StatusDetails": "Success",
            "OutputS3BucketName": "",
            "OutputS3KeyPrefix": "",
            "MaxConcurrency": "50",
            "MaxErrors": "100%",
            "TargetCount": 6,
            "CompletedCount": 6,
            "ErrorCount": 0,
            "DeliveryTimedOutCount": 0,
            "ServiceRole": "",
            "NotificationConfig": {
                "NotificationArn": "",
                "NotificationEvents": [],
```

```
                "NotificationType": ""
            },
            "CloudWatchOutputConfig": {
                "CloudWatchLogGroupName": "",
                "CloudWatchOutputEnabled": false
            }
        }
        {
            "CommandId": "e9ade581-c03d-476b-9b07-26667EXAMPLE",
            "DocumentName": "AWS-FindWindowsUpdates",
            "DocumentVersion": "1",
            "Comment": "",
            "ExpiresAfter": "2020-01-24T12:37:31.874000-08:00",
            "Parameters": {
                "KbArticleIds": [
                    ""
                ],
                "UpdateLevel": [
                    "All"
                ]
            },
            "InstanceIds": [],
            "Targets": [
                {
                    "Key": "InstanceIds",
                    "Values": [
                        "i-00ec29b21eEXAMPLE",
                        "i-09911ddd90EXAMPLE"
                    ]
                }
            ],
            "RequestedDateTime": "2020-01-24T11:27:31.874000-08:00",
            "Status": "Success",
            "StatusDetails": "Success",
            "OutputS3BucketName": "my-us-east-2-bucket",
            "OutputS3KeyPrefix": "my-rc-output",
            "MaxConcurrency": "50",
            "MaxErrors": "0",
            "TargetCount": 2,
            "CompletedCount": 2,
            "ErrorCount": 0,
            "DeliveryTimedOutCount": 0,
            "ServiceRole": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
```

```
            "NotificationConfig": {
                "NotificationArn": "arn:aws:sns:us-east-2:111222333444:my-us-
east-2-notification-arn",
                "NotificationEvents": [
                    "All"
                ],
                "NotificationType": "Invocation"
            },
            "CloudWatchOutputConfig": {
                "CloudWatchLogGroupName": "",
                "CloudWatchOutputEnabled": false
            }
        }
        {
            "CommandId": "d539b6c3-70e8-4853-80e5-0ce4fEXAMPLE",
            "DocumentName": "AWS-RunPatchBaseline",
            "DocumentVersion": "1",
            "Comment": "",
            "ExpiresAfter": "2020-01-24T12:21:04.350000-08:00",
            "Parameters": {
                "InstallOverrideList": [
                    ""
                ],
                "Operation": [
                    "Install"
                ],
                "RebootOption": [
                    "RebootIfNeeded"
                ],
                "SnapshotId": [
                    ""
                ]
            },
            "InstanceIds": [],
            "Targets": [
                {
                    "Key": "InstanceIds",
                    "Values": [
                        "i-00ec29b21eEXAMPLE",
                        "i-09911ddd90EXAMPLE"
                    ]
                }
            ],
            "RequestedDateTime": "2020-01-24T11:11:04.350000-08:00",
```

```
            "Status": "Success",
            "StatusDetails": "Success",
            "OutputS3BucketName": "my-us-east-2-bucket",
            "OutputS3KeyPrefix": "my-rc-output",
            "MaxConcurrency": "50",
            "MaxErrors": "0",
            "TargetCount": 2,
            "CompletedCount": 2,
            "ErrorCount": 0,
            "DeliveryTimedOutCount": 0,
            "ServiceRole": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
            "NotificationConfig": {
                "NotificationArn": "arn:aws:sns:us-east-2:111222333444:my-us-
east-2-notification-arn",
                "NotificationEvents": [
                    "All"
                ],
                "NotificationType": "Invocation"
            },
            "CloudWatchOutputConfig": {
                "CloudWatchLogGroupName": "",
                "CloudWatchOutputEnabled": false
            }
        }
    ]
}
```

Pour plus d'informations, voir Exécuter des commandes à l'aide de Systems Manager Run Command dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, voir ListCommandsla section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple répertorie toutes les commandes demandées.

```
Get-SSMCommand
```

Sortie :

```
CommandId          : 4b75a163-d39a-4d97-87c9-98ae52c6be35
Comment            : Apply association with id at update time: 4cc73e42-
d5ae-4879-84f8-57e09c0efcd0
CompletedCount     : 1
DocumentName       : AWS-RefreshAssociation
ErrorCount         : 0
ExpiresAfter       : 2/24/2017 3:19:08 AM
InstanceIds        : {i-0cb2b964d3e14fd9f}
MaxConcurrency     : 50
MaxErrors          : 0
NotificationConfig : Amazon.SimpleSystemsManagement.Model.NotificationConfig
OutputS3BucketName :
OutputS3KeyPrefix  :
OutputS3Region     :
Parameters         : {[associationIds,
 Amazon.Runtime.Internal.Util.AlwaysSendList`1[System.String]]}
RequestedDateTime  : 2/24/2017 3:18:08 AM
ServiceRole        :
Status             : Success
StatusDetails      : Success
TargetCount        : 1
Targets            : {}
```

Exemple 2 : Cet exemple obtient le statut d'une commande spécifique.

```
Get-SSMCommand -CommandId "4b75a163-d39a-4d97-87c9-98ae52c6be35"
```

Exemple 3 : Cet exemple récupère toutes les commandes SSM invoquées après 2019-04-01T 00:00:00 Z

```
Get-SSMCommand -Filter @{Key="InvokedAfter";Value="2019-04-01T00:00:00Z"} |
 Select-Object CommandId, DocumentName, Status, RequestedDateTime | Sort-Object -
Property RequestedDateTime -Descending
```

Sortie :

```
CommandId                            DocumentName              Status
 RequestedDateTime
---------                            ------------              ------
 -----------------
edb1b23e-456a-7adb-aef8-90e-012ac34f AWS-RunPowerShellScript    Cancelled
 4/16/2019 5:45:23 AM
```

```
1a2dc3fb-4567-890d-a1ad-234b5d6bc7d9 AWS-ConfigureAWSPackage    Success
 4/6/2019 9:19:42 AM
12c3456c-7e90-4f12-1232-1234f5b67893 KT-Retrieve-Cloud-Type-Win Failed
 4/2/2019 4:13:07 AM
fe123b45-240c-4123-a2b3-234bdd567ecf AWS-RunInspecChecks        Failed
 4/1/2019 2:27:31 PM
1eb23aa4-567d-4123-12a3-4c1c2ab34561 AWS-RunPowerShellScript    Success
 4/1/2019 1:05:55 PM
1c2f3bb4-ee12-4bc1-1a23-12345eea123e AWS-RunInspecChecks        Failed
 4/1/2019 11:13:09 AM
```

- Pour plus de détails sur l'API, consultez la section ListCommandsRéférence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`ListComplianceItems`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`ListComplianceItems`.

CLI

AWS CLI

Pour répertorier les éléments de conformité pour une instance spécifique

Cet exemple répertorie tous les éléments de conformité pour l'instance spécifiée.

Commande :

```
aws ssm list-compliance-items --resource-ids "i-1234567890abcdef0" --resource-
types "ManagedInstance"
```

Sortie :

```
{
  "ComplianceItems": [
      {
```

```
            "ComplianceType": "Association",
            "ResourceType": "ManagedInstance",
            "ResourceId": "i-1234567890abcdef0",
            "Id": "8dfe3659-4309-493a-8755-0123456789ab",
            "Title": "",
            "Status": "COMPLIANT",
            "Severity": "UNSPECIFIED",
            "ExecutionSummary": {
                "ExecutionTime": 1550408470.0
            },
            "Details": {
                "DocumentName": "AWS-GatherSoftwareInventory",
                "DocumentVersion": "1"
            }
        },
        {
            "ComplianceType": "Association",
            "ResourceType": "ManagedInstance",
            "ResourceId": "i-1234567890abcdef0",
            "Id": "e4c2ed6d-516f-41aa-aa2a-0123456789ab",
            "Title": "",
            "Status": "COMPLIANT",
            "Severity": "UNSPECIFIED",
            "ExecutionSummary": {
                "ExecutionTime": 1550508475.0
            },
            "Details": {
                "DocumentName": "AWS-UpdateSSMAgent",
                "DocumentVersion": "1"
            }
        },
                ...
    ],
    "NextToken": "--token string truncated--"
}
```

Pour répertorier les éléments de conformité pour une instance et un ID d'association spécifiques

Cet exemple répertorie tous les éléments de conformité pour l'instance et l'ID d'association spécifiés.

Commande :

```
aws ssm list-compliance-items --resource-ids "i-1234567890abcdef0" --resource-
types "ManagedInstance" --filters
 "Key=ComplianceType,Values=Association,Type=EQUAL"
 "Key=Id,Values=e4c2ed6d-516f-41aa-aa2a-0123456789ab,Type=EQUAL"
```

Pour répertorier les éléments de conformité d'une instance après une date et une heure spécifiques

Cet exemple répertorie tous les éléments de conformité d'une instance après la date et l'heure spécifiées.

Commande :

```
aws ssm list-compliance-items --resource-ids "i-1234567890abcdef0" --resource-
types "ManagedInstance" --filters
 "Key=ExecutionTime,Values=2019-02-18T16:00:00Z,Type=GREATER_THAN"
```

- Pour plus de détails sur l'API, consultez la section ListComplianceÉléments du manuel de référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple répertorie la liste des éléments de conformité pour l'identifiant et le type de ressource donnés, le type de conformité de filtrage étant « Association »

```
Get-SSMComplianceItemList -ResourceId i-1a2caf345f67d0dc2 -ResourceType
 ManagedInstance -Filter @{Key="ComplianceType";Values="Association"}
```

Sortie :

```
ComplianceType   : Association
Details          : {[DocumentName, AWS-GatherSoftwareInventory],
 [DocumentVersion, 1]}
ExecutionSummary :
 Amazon.SimpleSystemsManagement.Model.ComplianceExecutionSummary
Id               : 123a45a1-c234-1234-1245-67891236db4e
ResourceId       : i-1a2caf345f67d0dc2
ResourceType     : ManagedInstance
Severity         : UNSPECIFIED
```

```
Status           : COMPLIANT
Title            :
```

- Pour plus de détails sur l'API, consultez la section ListComplianceRéférence des éléments des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`ListComplianceSummaries`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`ListComplianceSummaries`.

CLI

AWS CLI

Pour répertorier les résumés de conformité pour tous les types de conformité

Cet exemple répertorie les résumés de conformité pour tous les types de conformité de votre compte.

Commande :

```
aws ssm list-compliance-summaries
```

Sortie :

```
{
  "ComplianceSummaryItems": [
      {
          "ComplianceType": "Association",
          "CompliantSummary": {
              "CompliantCount": 2,
              "SeveritySummary": {
                  "CriticalCount": 0,
                  "HighCount": 0,
                  "MediumCount": 0,
                  "LowCount": 0,
                  "InformationalCount": 0,
                  "UnspecifiedCount": 2
```

```
                }
            },
            "NonCompliantSummary": {
                "NonCompliantCount": 0,
                "SeveritySummary": {
                    "CriticalCount": 0,
                    "HighCount": 0,
                    "MediumCount": 0,
                    "LowCount": 0,
                    "InformationalCount": 0,
                    "UnspecifiedCount": 0
                }
            }
        },
        {
            "ComplianceType": "Patch",
            "CompliantSummary": {
                "CompliantCount": 1,
                "SeveritySummary": {
                    "CriticalCount": 0,
                    "HighCount": 0,
                    "MediumCount": 0,
                    "LowCount": 0,
                    "InformationalCount": 0,
                    "UnspecifiedCount": 1
                }
            },
            "NonCompliantSummary": {
                "NonCompliantCount": 1,
                "SeveritySummary": {
                    "CriticalCount": 1,
                    "HighCount": 0,
                    "MediumCount": 0,
                    "LowCount": 0,
                    "InformationalCount": 0,
                    "UnspecifiedCount": 0
                }
            }
        },
                ...
    ],
    "NextToken": "eyJOZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAyfQ=="
}
```

Pour répertorier les résumés de conformité pour un type de conformité spécifique

Cet exemple répertorie le résumé de conformité pour le type de conformité Patch.

Commande :

```
aws ssm list-compliance-summaries --filters
 "Key=ComplianceType,Values=Patch,Type=EQUAL"
```

- Pour plus de détails sur l'API, voir ListComplianceRésumés dans AWS CLI Command Reference.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple renvoie un décompte récapitulatif des ressources conformes et non conformes pour tous les types de conformité.

```
Get-SSMComplianceSummaryList
```

Sortie :

```
ComplianceType CompliantSummary
 NonCompliantSummary
-------------- ----------------
 -------------------
FleetTotal     Amazon.SimpleSystemsManagement.Model.CompliantSummary
 Amazon.SimpleSystemsManagement.Model.NonCompliantSummary
Association    Amazon.SimpleSystemsManagement.Model.CompliantSummary
 Amazon.SimpleSystemsManagement.Model.NonCompliantSummary
Custom:InSpec  Amazon.SimpleSystemsManagement.Model.CompliantSummary
 Amazon.SimpleSystemsManagement.Model.NonCompliantSummary
Patch          Amazon.SimpleSystemsManagement.Model.CompliantSummary
 Amazon.SimpleSystemsManagement.Model.NonCompliantSummary
```

- Pour plus de détails sur l'API, consultez la section ListComplianceRésumés dans la référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`ListDocumentVersions`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`ListDocumentVersions`.

CLI

AWS CLI

Pour répertorier les versions de documents

L'`list-document-versions`exemple suivant répertorie toutes les versions d'un document Systems Manager.

```
aws ssm list-document-versions \
    --name "Example"
```

Sortie :

```
{
    "DocumentVersions": [
        {
            "Name": "Example",
            "DocumentVersion": "1",
            "CreatedDate": 1583257938.266,
            "IsDefaultVersion": true,
            "DocumentFormat": "YAML",
            "Status": "Active"
        }
    ]
}
```

Pour plus d'informations, consultez la section Envoi de commandes utilisant le paramètre de version du document dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, voir ListDocumentVersions dans AWS CLI Command Reference.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple renvoie la liste des autorisations pour un document.

```
Get-SSMDocumentPermission -Name "RunShellScript" -PermissionType "Share"
```

Sortie :

```
all
```

- Pour plus de détails sur l'API, consultez la section ListDocumentVersions dans la référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`ListDocuments`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`ListDocuments`.

CLI

AWS CLI

Exemple 1 : pour répertorier des documents

L'`list-documents`exemple suivant répertorie les documents appartenant au compte demandeur étiquetés avec la balise personnalisée.

```
aws ssm list-documents \
    --filters Key=Owner,Values=Self Key=tag:DocUse,Values=Testing
```

Sortie :

```
{
    "DocumentIdentifiers": [
        {
            "Name": "Example",
```

```
            "Owner": "29884EXAMPLE",
            "PlatformTypes": [
                "Windows",
                "Linux"
            ],
            "DocumentVersion": "1",
            "DocumentType": "Automation",
            "SchemaVersion": "0.3",
            "DocumentFormat": "YAML",
            "Tags": [
                {
                    "Key": "DocUse",
                    "Value": "Testing"
                }
            ]
        }
    ]
}
```

Pour plus d'informations, consultez les documents de AWS Systems Manager dans le guide de l'utilisateur de AWS Systems Manager.

Exemple 2 : pour répertorier les documents partagés

L'`list-documents`exemple suivant répertorie les documents partagés, y compris les documents partagés privés qui ne sont pas détenus par AWS.

```
aws ssm list-documents \
    --filters Key=Name,Values=sharedDocNamePrefix  Key=Owner,Values=Private
```

Sortie :

```
{
    "DocumentIdentifiers": [
        {
            "Name": "Example",
            "Owner": "12345EXAMPLE",
            "PlatformTypes": [
                "Windows",
                "Linux"
            ],
            "DocumentVersion": "1",
            "DocumentType": "Command",
```

```
            "SchemaVersion": "0.3",
            "DocumentFormat": "YAML",
            "Tags": []
        }
    ]
}
```

Pour plus d'informations, consultez les documents de AWS Systems Manager dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous ListDocumentsà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Répertorie tous les documents de configuration de votre compte.

```
Get-SSMDocumentList
```

Sortie :

```
DocumentType    : Command
DocumentVersion : 1
Name            : AWS-ApplyPatchBaseline
Owner           : Amazon
PlatformTypes   : {Windows}
SchemaVersion   : 1.2

DocumentType    : Command
DocumentVersion : 1
Name            : AWS-ConfigureAWSPackage
Owner           : Amazon
PlatformTypes   : {Windows, Linux}
SchemaVersion   : 2.0

DocumentType    : Command
DocumentVersion : 1
Name            : AWS-ConfigureCloudWatch
Owner           : Amazon
PlatformTypes   : {Windows}
```

```
SchemaVersion   : 1.2
...
```

Exemple 2 : Cet exemple récupère tous les documents d'automatisation dont le nom correspond à « Platform »

```
Get-SSMDocumentList -DocumentFilterList @{Key="DocumentType";Value="Automation"}
 | Where-Object Name -Match "Platform"
```

Sortie :

```
DocumentFormat  : JSON
DocumentType    : Automation
DocumentVersion : 7
Name            : KT-Get-Platform
Owner           : 987654123456
PlatformTypes   : {Windows, Linux}
SchemaVersion   : 0.3
Tags            : {}
TargetType      :
VersionName     :
```

- Pour plus de détails sur l'API, reportez-vous ListDocumentsà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`ListInventoryEntries`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`ListInventoryEntries`.

CLI

AWS CLI

Exemple 1 : pour afficher des entrées de type d'inventaire spécifiques pour une instance

L'`list-inventory-entries`exemple suivant répertorie les entrées d'inventaire pour le type:Application AWS inventory sur une instance spécifique.

```
aws ssm list-inventory-entries \
    --instance-id "i-1234567890abcdef0" \
    --type-name "AWS:Application"
```

Sortie :

```
{
  "TypeName": "AWS:Application",
  "InstanceId": "i-1234567890abcdef0",
  "SchemaVersion": "1.1",
  "CaptureTime": "2019-02-15T12:17:55Z",
  "Entries": [
    {
      "Architecture": "i386",
      "Name": "Amazon SSM Agent",
      "PackageId": "{88a60be2-89a1-4df8-812a-80863c2a2b68}",
      "Publisher": "Amazon Web Services",
      "Version": "2.3.274.0"
    },
    {
      "Architecture": "x86_64",
      "InstalledTime": "2018-05-03T13:42:34Z",
      "Name": "AmazonCloudWatchAgent",
      "Publisher": "",
      "Version": "1.200442.0"
    }
  ]
}
```

Exemple 2 : pour afficher les entrées d'inventaire personnalisées attribuées à une instance

L'`list-inventory-entries`exemple suivant répertorie une entrée d'inventaire personnalisée attribuée à une instance.

```
aws ssm list-inventory-entries \
    --instance-id "i-1234567890abcdef0" \
    --type-name "Custom:RackInfo"
```

Sortie :

```
{
  "TypeName": "Custom:RackInfo",
```

```
  "InstanceId": "i-1234567890abcdef0",
  "SchemaVersion": "1.0",
  "CaptureTime": "2021-05-22T10:01:01Z",
  "Entries": [
    {
      "RackLocation": "Bay B/Row C/Rack D/Shelf E"
    }
  ]
}
```

- Pour plus de détails sur l'API, consultez la section ListInventoryEntrées dans AWS CLI la référence des commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple répertorie toutes les entrées d'inventaire personnalisées pour une instance.

```
Get-SSMInventoryEntriesList -InstanceId "i-0cb2b964d3e14fd9f" -TypeName
 "Custom:RackInfo"
```

Sortie :

```
CaptureTime   : 2016-08-22T10:01:01Z
Entries       :
 {Amazon.Runtime.Internal.Util.AlwaysSendDictionary`2[System.String,System.String]}
InstanceId    : i-0cb2b964d3e14fd9f
NextToken     :
SchemaVersion : 1.0
TypeName      : Custom:RackInfo
```

Exemple 2 : Cet exemple répertorie les détails.

```
(Get-SSMInventoryEntriesList -InstanceId "i-0cb2b964d3e14fd9f" -TypeName
 "Custom:RackInfo").Entries
```

Sortie :

```
Key          Value
```

```
---          -----
RackLocation Bay B/Row C/Rack D/Shelf E
```

- Pour plus de détails sur l'API, consultez la section ListInventoryEntrées dans la référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`ListResourceComplianceSummaries`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`ListResourceComplianceSummaries`.

CLI

AWS CLI

Pour répertorier le nombre récapitulatif de conformité au niveau des ressources

Cet exemple répertorie les comptes récapitulatifs de conformité au niveau des ressources.

Commande :

```
aws ssm list-resource-compliance-summaries
```

Sortie :

```
{
  "ResourceComplianceSummaryItems": [
      {
          "ComplianceType": "Association",
          "ResourceType": "ManagedInstance",
          "ResourceId": "i-1234567890abcdef0",
          "Status": "COMPLIANT",
          "OverallSeverity": "UNSPECIFIED",
          "ExecutionSummary": {
              "ExecutionTime": 1550509273.0
          },
          "CompliantSummary": {
```

```
            "CompliantCount": 2,
            "SeveritySummary": {
                "CriticalCount": 0,
                "HighCount": 0,
                "MediumCount": 0,
                "LowCount": 0,
                "InformationalCount": 0,
                "UnspecifiedCount": 2
            }
        },
        "NonCompliantSummary": {
            "NonCompliantCount": 0,
            "SeveritySummary": {
                "CriticalCount": 0,
                "HighCount": 0,
                "MediumCount": 0,
                "LowCount": 0,
                "InformationalCount": 0,
                "UnspecifiedCount": 0
            }
        }
    },
    {
        "ComplianceType": "Patch",
        "ResourceType": "ManagedInstance",
        "ResourceId": "i-9876543210abcdef0",
        "Status": "COMPLIANT",
        "OverallSeverity": "UNSPECIFIED",
        "ExecutionSummary": {
            "ExecutionTime": 1550248550.0,
            "ExecutionId": "7abb6378-a4a5-4f10-8312-0123456789ab",
            "ExecutionType": "Command"
        },
        "CompliantSummary": {
            "CompliantCount": 397,
            "SeveritySummary": {
                "CriticalCount": 0,
                "HighCount": 0,
                "MediumCount": 0,
                "LowCount": 0,
                "InformationalCount": 0,
                "UnspecifiedCount": 397
            }
        },
```

```
            "NonCompliantSummary": {
                "NonCompliantCount": 0,
                "SeveritySummary": {
                    "CriticalCount": 0,
                    "HighCount": 0,
                    "MediumCount": 0,
                    "LowCount": 0,
                    "InformationalCount": 0,
                    "UnspecifiedCount": 0
                }
            }
        }
    ],
    "NextToken": "--token string truncated--"
}
```

Pour répertorier les résumés de conformité au niveau des ressources pour un type de conformité spécifique

Cet exemple répertorie les résumés de conformité au niveau des ressources pour le type de conformité Patch.

Commande :

```
aws ssm list-resource-compliance-summaries --filters
 "Key=ComplianceType,Values=Patch,Type=EQUAL"
```

- Pour plus de détails sur l'API, reportez-vous ListResourceComplianceSummariesà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple obtient un décompte récapitulatif au niveau des ressources. Le résumé inclut des informations sur les statuts de conformité et de non-conformité ainsi que le nombre détaillé de gravité des éléments de conformité pour les produits correspondant à « Windows10 ». Comme la valeur MaxResult par défaut est 100 si le paramètre n'est pas spécifié et que cette valeur n'est pas valide, le MaxResult paramètre est ajouté et la valeur est définie sur 50.

```
$FilterValues = @{
  "Key"="Product"
        "Type"="EQUAL"
        "Values"="Windows10"
}
        Get-SSMResourceComplianceSummaryList -Filter $FilterValues -MaxResult 50
```

- Pour plus de détails sur l'API, reportez-vous ListResourceComplianceSummariesà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`ListTagsForResource`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`ListTagsForResource`.

CLI

AWS CLI

Pour répertorier les balises appliquées à une ligne de base de correctifs

L'`list-tags-for-resource`exemple suivant répertorie les balises d'une ligne de base de correctifs.

```
aws ssm list-tags-for-resource \
    --resource-type "PatchBaseline" \
    --resource-id "pb-0123456789abcdef0"
```

Sortie :

```
{
    "TagList": [
        {
            "Key": "Environment",
            "Value": "Production"
        },
        {
```

```
            "Key": "Region",
            "Value": "EMEA"
        }
    ]
}
```

Pour plus d'informations, consultez la section AWS Ressources de balisage dans le manuel de référence AWS général.

- Pour plus de détails sur l'API, reportez-vous ListTagsForResourceà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple répertorie les balises d'une fenêtre de maintenance.

```
Get-SSMResourceTag -ResourceId "mw-03eb9db42890fb82d" -ResourceType
 "MaintenanceWindow"
```

Sortie :

```
Key   Value
---   -----
Stack Production
```

- Pour plus de détails sur l'API, reportez-vous ListTagsForResourceà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`ModifyDocumentPermission`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`ModifyDocumentPermission`.

CLI

AWS CLI

Pour modifier les autorisations relatives aux documents

L'`modify-document-permission`exemple suivant partage publiquement un document Systems Manager.

```
aws ssm modify-document-permission \
    --name "Example" \
    --permission-type "Share" \
    --account-ids-to-add "All"
```

Cette commande ne produit aucun résultat.

Pour plus d'informations, consultez la section Partager un document Systems Manager dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, consultez la section ModifyDocumentAutorisation dans AWS CLI la référence des commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple ajoute des autorisations de « partage » à tous les comptes associés à un document. Il n'y a aucune sortie si la commande aboutit.

```
Edit-SSMDocumentPermission -Name "RunShellScript" -PermissionType "Share" -
AccountIdsToAdd all
```

Exemple 2 : Cet exemple ajoute des autorisations de « partage » à un compte spécifique pour un document. Il n'y a aucune sortie si la commande aboutit.

```
Edit-SSMDocumentPermission -Name "RunShellScriptNew" -PermissionType "Share" -
AccountIdsToAdd "123456789012"
```

- Pour plus de détails sur l'API, consultez la section ModifyDocumentAutorisation dans la référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`PutComplianceItems`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`PutComplianceItems`.

CLI

AWS CLI

Pour enregistrer un type de conformité et des informations de conformité sur une instance désignée

Cet exemple enregistre le type `Custom:AVCheck` de conformité sur l'instance gérée spécifiée. Il n'y a pas de sortie si la commande réussit.

Commande :

```
aws ssm put-compliance-items --resource-id "i-1234567890abcdef0" --
resource-type "ManagedInstance" --compliance-type "Custom:AVCheck"
 --execution-summary "ExecutionTime=2019-02-18T16:00:00Z" --items
 "Id=Version2.0,Title=ScanHost,Severity=CRITICAL,Status=COMPLIANT"
```

- Pour plus de détails sur l'API, consultez la section PutComplianceÉléments du manuel de référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple écrit un élément de conformité personnalisé pour l'instance gérée donnée

```
$item = [Amazon.SimpleSystemsManagement.Model.ComplianceItemEntry]::new()
$item.Id = "07Jun2019-3"
$item.Severity="LOW"
$item.Status="COMPLIANT"
$item.Title="Fin-test-1 - custom"
```

```
Write-SSMComplianceItem -ResourceId mi-012dcb3ecea45b678 -ComplianceType
 Custom:VSSCompliant2 -ResourceType ManagedInstance -Item $item -
ExecutionSummary_ExecutionTime "07-Jun-2019"
```

- Pour plus de détails sur l'API, consultez la section PutComplianceRéférence des éléments des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`PutInventory`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`PutInventory`.

CLI

AWS CLI

Pour attribuer des métadonnées client à une instance

Cet exemple attribue les informations sur les emplacements des racks à une instance. Il n'y a pas de sortie si la commande réussit.

Commande (Linux) :

```
aws ssm put-inventory --instance-id "i-016648b75dd622dab" --items
 '[{"TypeName": "Custom:RackInfo","SchemaVersion": "1.0","CaptureTime":
 "2019-01-22T10:01:01Z","Content":[{"RackLocation": "Bay B/Row C/Rack D/Shelf
 E"}]}]'
```

Commande (Windows) :

```
aws ssm put-inventory --instance-id "i-016648b75dd622dab" --items
 "TypeName=Custom:RackInfo,SchemaVersion=1.0,CaptureTime=2019-01-22T10:01:01Z,Content=[{R
 B/Row C/Rack D/Shelf F'}]"
```

- Pour plus de détails sur l'API, reportez-vous PutInventoryà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple attribue des informations sur l'emplacement du rack à une instance. Il n'y a aucune sortie si la commande aboutit.

```
$data = New-Object
 "System.Collections.Generic.Dictionary[System.String,System.String]"
$data.Add("RackLocation", "Bay B/Row C/Rack D/Shelf F")

$items = New-Object
 "System.Collections.Generic.List[System.Collections.Generic.Dictionary[System.String,
 System.String]]"
$items.Add($data)

$customInventoryItem = New-Object
 Amazon.SimpleSystemsManagement.Model.InventoryItem
$customInventoryItem.CaptureTime = "2016-08-22T10:01:01Z"
$customInventoryItem.Content = $items
$customInventoryItem.TypeName = "Custom:TestRackInfo2"
$customInventoryItem.SchemaVersion = "1.0"

$inventoryItems = @($customInventoryItem)

Write-SSMInventory -InstanceId "i-0cb2b964d3e14fd9f" -Item $inventoryItems
```

- Pour plus de détails sur l'API, reportez-vous PutInventoryà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`PutParameter`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`PutParameter`.

CLI

AWS CLI

Exemple 1 : pour modifier la valeur d'un paramètre

L'`put-parameter`exemple suivant modifie la valeur du paramètre spécifié.

```
aws ssm put-parameter \
    --name "MyStringParameter" \
    --type "String" \
    --value "Vici" \
    --overwrite
```

Sortie :

```
{
    "Version": 2,
    "Tier": "Standard"
}
```

Pour plus d'informations, consultez Create a Systems Manager parameter (AWS CLI), « Managing parameter tiers"__ et Working with parameter policies dans le guide de l'utilisateur de Systems AWS Manager. < https://docs.aws.amazon.com/systems-manager/latest/userguide/parameter-store-advanced-parameters.html>

Exemple 2 : pour créer un paramètre avancé

L'`put-parameter`exemple suivant crée un paramètre avancé.

```
aws ssm put-parameter \
    --name "MyAdvancedParameter" \
    --description "This is an advanced parameter" \
    --value "Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do
 eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim
 veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo
 consequat [truncated]" \
    --type "String" \
    --tier Advanced
```

Sortie :

```
{
    "Version": 1,
    "Tier": "Advanced"
}
```

Pour plus d'informations, consultez Create a Systems Manager parameter (AWS CLI), « Managing parameter tiers"__ et Working with parameter policies dans le guide de l'utilisateur de Systems AWS Manager. < https://docs.aws.amazon.com/systems-manager/latest/userguide/parameter-store-advanced-parameters.html>

Exemple 3 : pour convertir un paramètre standard en paramètre avancé

L'`put-parameter`exemple suivant convertit un paramètre standard existant en paramètre avancé.

```
aws ssm put-parameter \
    --name "MyConvertedParameter" \
    --value "abc123" \
    --type "String" \
    --tier Advanced \
    --overwrite
```

Sortie :

```
{
    "Version": 2,
    "Tier": "Advanced"
}
```

Pour plus d'informations, consultez Create a Systems Manager parameter (AWS CLI), « Managing parameter tiers"__ et Working with parameter policies dans le guide de l'utilisateur de Systems AWS Manager. < https://docs.aws.amazon.com/systems-manager/latest/userguide/parameter-store-advanced-parameters.html>

Exemple 4 : pour créer un paramètre auquel est attachée une politique

L'`put-parameter`exemple suivant crée un paramètre avancé auquel est attachée une politique de paramètres.

```
aws ssm put-parameter \
```

```
    --name "/Finance/Payroll/q2accesskey" \
    --value "P@sSwW)rd" \
    --type "SecureString" \
    --tier Advanced \
    --policies "[{\"Type\":\"Expiration\",\"Version\":\"1.0\",\"Attributes\":
{\"Timestamp\":\"2020-06-30T00:00:00.000Z\"}},{\"Type\":\"ExpirationNotification
\",\"Version\":\"1.0\",\"Attributes\":{\"Before\":\"5\",\"Unit\":\"Days\"}},
{\"Type\":\"NoChangeNotification\",\"Version\":\"1.0\",\"Attributes\":{\"After\":
\"60\",\"Unit\":\"Days\"}}]"
```

Sortie :

```
{
    "Version": 1,
    "Tier": "Advanced"
}
```

Pour plus d'informations, consultez Create a Systems Manager parameter (AWS CLI), « Managing parameter tiers"__ et Working with parameter policies dans le guide de l'utilisateur de Systems AWS Manager. < https://docs.aws.amazon.com/systems-manager/latest/userguide/parameter-store-advanced-parameters.html>

Exemple 5 : pour ajouter une politique à un paramètre existant

L'`put-parameter`exemple suivant associe une politique à un paramètre avancé existant.

```
aws ssm put-parameter \
    --name "/Finance/Payroll/q2accesskey" \
    --value "N3wP@sSwW)rd" \
    --type "SecureString" \
    --tier Advanced \
    --policies "[{\"Type\":\"Expiration\",\"Version\":\"1.0\",\"Attributes\":
{\"Timestamp\":\"2020-06-30T00:00:00.000Z\"}},{\"Type\":\"ExpirationNotification
\",\"Version\":\"1.0\",\"Attributes\":{\"Before\":\"5\",\"Unit\":\"Days\"}},
{\"Type\":\"NoChangeNotification\",\"Version\":\"1.0\",\"Attributes\":{\"After\":
\"60\",\"Unit\":\"Days\"}}]"
    --overwrite
```

Sortie :

```
{
    "Version": 2,
```

```
    "Tier": "Advanced"
}
```

Pour plus d'informations, consultez Create a Systems Manager parameter (AWS CLI), « Managing parameter tiers"__ et Working with parameter policies dans le guide de l'utilisateur de Systems AWS Manager. < https://docs.aws.amazon.com/systems-manager/latest/userguide/parameter-store-advanced-parameters.html>

- Pour plus de détails sur l'API, reportez-vous PutParameterà la section Référence des AWS CLI commandes.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.ssm.SsmClient;
import software.amazon.awssdk.services.ssm.model.ParameterType;
import software.amazon.awssdk.services.ssm.model.PutParameterRequest;
import software.amazon.awssdk.services.ssm.model.SsmException;

public class PutParameter {

    public static void main(String[] args) {
        final String usage = """

                Usage:
                    <paraName>

                Where:
                    paraName - The name of the parameter.
                    paraValue - The value of the parameter.
                """;

        if (args.length != 2) {
```

```
            System.out.println(usage);
            System.exit(1);
        }

        String paraName = args[0];
        String paraValue = args[1];
        Region region = Region.US_EAST_1;
        SsmClient ssmClient = SsmClient.builder()
                .region(region)
                .build();

        putParaValue(ssmClient, paraName, paraValue);
        ssmClient.close();
    }

    public static void putParaValue(SsmClient ssmClient, String paraName, String
 value) {
        try {
            PutParameterRequest parameterRequest = PutParameterRequest.builder()
                    .name(paraName)
                    .type(ParameterType.STRING)
                    .value(value)
                    .build();

            ssmClient.putParameter(parameterRequest);
            System.out.println("The parameter was successfully added.");

        } catch (SsmException e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }
}
```

- Pour plus de détails sur l'API, reportez-vous PutParameterà la section Référence des AWS SDK for Java 2.x API.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple crée un paramètre. Il n'y a aucune sortie si la commande aboutit.

```
Write-SSMParameter -Name "Welcome" -Type "String" -Value "helloWorld"
```

Exemple 2 : Cet exemple modifie un paramètre. Il n'y a aucune sortie si la commande aboutit.

```
Write-SSMParameter -Name "Welcome" -Type "String" -Value "Good day, Sunshine!" -
Overwrite $true
```

- Pour plus de détails sur l'API, reportez-vous PutParameterà la section Référence des AWS Tools for PowerShell applets de commande.

Rust

SDK pour Rust

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
async fn make_parameter(
    client: &Client,
    name: &str,
    value: &str,
    description: &str,
) -> Result<(), Error> {
    let resp = client
        .put_parameter()
        .overwrite(true)
        .r#type(ParameterType::String)
        .name(name)
        .value(value)
        .description(description)
        .send()
        .await?;

    println!("Success! Parameter now has version: {}", resp.version());

    Ok(())
}
```

- Pour plus de détails sur l'API, voir PutParameterla section de référence de l'API AWS SDK for Rust.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`RegisterDefaultPatchBaseline`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`RegisterDefaultPatchBaseline`.

CLI

AWS CLI

Pour définir la ligne de base de correctif par défaut

L'`register-default-patch-baseline`exemple suivant enregistre la ligne de base de correctif personnalisée spécifiée comme ligne de base de correctif par défaut pour le type de système d'exploitation qu'elle prend en charge.

```
aws ssm register-default-patch-baseline \
    --baseline-id "pb-abc123cf9bEXAMPLE"
```

Sortie :

```
{
    "BaselineId":"pb-abc123cf9bEXAMPLE"
}
```

L'`register-default-patch-baseline`exemple suivant enregistre la ligne de base de correctifs par défaut fournie par AWS CentOS comme ligne de base de correctifs par défaut.

```
aws ssm register-default-patch-baseline \
    --baseline-id "arn:aws:ssm:us-east-2:733109147000:patchbaseline/
pb-0574b43a65ea646ed"
```

Sortie :

```
{
    "BaselineId":"pb-abc123cf9bEXAMPLE"
}
```

Pour plus d'informations, reportez-vous à la section À propos des lignes de base de correctifs prédéfinies et personnalisées dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous RegisterDefaultPatchBaselineà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple enregistre une ligne de base de correctif comme ligne de base de correctif par défaut.

```
Register-SSMDefaultPatchBaseline -BaselineId "pb-03da896ca3b68b639"
```

Sortie :

```
pb-03da896ca3b68b639
```

- Pour plus de détails sur l'API, reportez-vous RegisterDefaultPatchBaselineà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`RegisterPatchBaselineForPatchGroup`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`RegisterPatchBaselineForPatchGroup`.

CLI

AWS CLI

Pour enregistrer une ligne de base de correctifs pour un groupe de correctifs

L'`register-patch-baseline-for-patch-group`exemple suivant enregistre une ligne de base de correctifs pour un groupe de correctifs.

```
aws ssm register-patch-baseline-for-patch-group \
    --baseline-id "pb-045f10b4f382baeda" \
    --patch-group "Production"
```

Sortie :

```
{
    "BaselineId": "pb-045f10b4f382baeda",
    "PatchGroup": "Production"
}
```

Pour plus d'informations, consultez les sections Create a Patch Group__ < https://docs.aws.amazon.com/systems-manager/latest/userguide/sysman-patch-group-tagging.html> et Add a Patch Group to a Patch Baseline dans le Guide de l'utilisateur de Systems AWS Manager.

- Pour plus de détails sur l'API, reportez-vous RegisterPatchBaselineForPatchGroupà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple enregistre une ligne de base de correctifs pour un groupe de correctifs.

```
Register-SSMPatchBaselineForPatchGroup -BaselineId "pb-03da896ca3b68b639" -
PatchGroup "Production"
```

Sortie :

```
BaselineId           PatchGroup
```

```
----------           ----------
pb-03da896ca3b68b639 Production
```

- Pour plus de détails sur l'API, reportez-vous RegisterPatchBaselineForPatchGroupà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`RegisterTargetWithMaintenanceWindow`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`RegisterTargetWithMaintenanceWindow`.

CLI

AWS CLI

Exemple 1 : pour enregistrer une cible unique avec une fenêtre de maintenance

L'`register-target-with-maintenance-window`exemple suivant enregistre une instance avec une fenêtre de maintenance.

```
aws ssm register-target-with-maintenance-window \
    --window-id "mw-ab12cd34ef56gh78" \
    --target "Key=InstanceIds,Values=i-0000293ffd8c57862" \
    --owner-information "Single instance" \
    --resource-type "INSTANCE"
```

Sortie :

```
{
    "WindowTargetId":"1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
}
```

Exemple 2 : pour enregistrer plusieurs cibles avec une fenêtre de maintenance à l'aide des ID d'instance

L'`register-target-with-maintenance-window`exemple suivant enregistre deux instances avec une fenêtre de maintenance en spécifiant leurs ID d'instance.

```
aws ssm register-target-with-maintenance-window \
    --window-id "mw-ab12cd34ef56gh78" \
    --target "Key=InstanceIds,Values=i-0000293ffd8c57862,i-0cb2b964d3e14fd9f" \
    --owner-information "Two instances in a list" \
    --resource-type "INSTANCE"
```

Sortie :

```
{
    "WindowTargetId":"1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
}
```

Exemple 3 : pour enregistrer des cibles avec une fenêtre de maintenance à l'aide de balises de ressources

L'`register-target-with-maintenance-window`exemple suivant enregistre les instances avec une fenêtre de maintenance en spécifiant les balises de ressources qui ont été appliquées aux instances.

```
aws ssm register-target-with-maintenance-window \
    --window-id "mw-06cf17cbefcb4bf4f" \
    --targets "Key=tag:Environment,Values=Prod" "Key=Role,Values=Web" \
    --owner-information "Production Web Servers" \
    --resource-type "INSTANCE"
```

Sortie :

```
{
    "WindowTargetId":"1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
}
```

Exemple 4 : pour enregistrer des cibles à l'aide d'un groupe de clés de balise

L'`register-target-with-maintenance-window`exemple suivant enregistre des instances auxquelles une ou plusieurs clés de balise leur sont attribuées, quelles que soient leurs valeurs clés.

```
aws ssm register-target-with-maintenance-window \
```

```
    --window-id "mw-0c50858d01EXAMPLE" \
    --resource-type "INSTANCE" \
    --target "Key=tag-key,Values=Name,Instance-Type,CostCenter"
```

Sortie :

```
{
    "WindowTargetId":"1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
}
```

Exemple 5 : pour enregistrer des cibles à l'aide d'un nom de groupe de ressources

L'`register-target-with-maintenance-window`exemple suivant enregistre un groupe de ressources spécifié, quel que soit le type de ressources qu'il contient.

```
aws ssm register-target-with-maintenance-window \
    --window-id "mw-0c50858d01EXAMPLE" \
    --resource-type "RESOURCE_GROUP" \
    --target "Key=resource-groups:Name,Values=MyResourceGroup"
```

Sortie :

```
{
    "WindowTargetId":"1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
}
```

Pour plus d'informations, consultez la section Enregistrer une instance cible avec la fenêtre de maintenance (AWS CLI) dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, voir RegisterTargetWithMaintenanceWindow in AWS CLI Command Reference.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple enregistre une instance avec une fenêtre de maintenance.

```
$option1 = @{Key="InstanceIds";Values=@("i-0000293ffd8c57862")}
Register-SSMTargetWithMaintenanceWindow -WindowId "mw-06cf17cbefcb4bf4f" -Target
 $option1 -OwnerInformation "Single instance" -ResourceType "INSTANCE"
```

Sortie :

```
d8e47760-23ed-46a5-9f28-927337725398
```

Exemple 2 : Cet exemple enregistre plusieurs instances avec une fenêtre de maintenance.

```
$option1 =
 @{Key="InstanceIds";Values=@("i-0000293ffd8c57862","i-0cb2b964d3e14fd9f")}
Register-SSMTargetWithMaintenanceWindow -WindowId "mw-06cf17cbefcb4bf4f" -Target
 $option1 -OwnerInformation "Single instance" -ResourceType "INSTANCE"
```

Sortie :

```
6ab5c208-9fc4-4697-84b7-b02a6cc25f7d
```

Exemple 3 : Cet exemple enregistre une instance avec une fenêtre de maintenance à l'aide de balises EC2.

```
$option1 = @{Key="tag:Environment";Values=@("Production")}
Register-SSMTargetWithMaintenanceWindow -WindowId "mw-06cf17cbefcb4bf4f" -Target
 $option1 -OwnerInformation "Production Web Servers" -ResourceType "INSTANCE"
```

Sortie :

```
2994977e-aefb-4a71-beac-df620352f184
```

- Pour plus de détails sur l'API, consultez RegisterTargetWithMaintenancela section Window in AWS Tools for PowerShell Cmdlet Reference.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`RegisterTaskWithMaintenanceWindow`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`RegisterTaskWithMaintenanceWindow`.

CLI

AWS CLI

Exemple 1 : pour enregistrer une tâche d'automatisation avec une fenêtre de maintenance

L'`register-task-with-maintenance-window`exemple suivant enregistre une tâche d'automatisation avec une fenêtre de maintenance ciblée sur une instance.

```
aws ssm register-task-with-maintenance-window \
    --window-id "mw-082dcd7649EXAMPLE" \
    --targets Key=InstanceIds,Values=i-1234520122EXAMPLE \
    --task-arn AWS-RestartEC2Instance \
    --service-role-arn arn:aws:iam::111222333444:role/SSM --task-type AUTOMATION
 \
    --task-invocation-parameters "{\"Automation\":{\"DocumentVersion\":\"\$LATEST
\",\"Parameters\":{\"InstanceId\":[\"{{RESOURCE_ID}}\"]}}}" \
    --priority 0 \
    --max-concurrency 1 \
    --max-errors 1 \
    --name "AutomationExample" \
    --description "Restarting EC2 Instance for maintenance"
```

Sortie :

```
{
    "WindowTaskId":"11144444-5555-6666-7777-88888888"
}
```

Pour plus d'informations, consultez la section Enregistrer une tâche avec la fenêtre de maintenance (AWS CLI) dans le guide de l'utilisateur de AWS Systems Manager.

Exemple 2 : pour enregistrer une tâche Lambda avec une fenêtre de maintenance

L'`register-task-with-maintenance-window`exemple suivant enregistre une tâche Lambda avec une fenêtre de maintenance ciblée sur une instance.

```
aws ssm register-task-with-maintenance-window \
    --window-id "mw-082dcd7649dee04e4" \
    --targets Key=InstanceIds,Values=i-12344d305eEXAMPLE \
    --task-arn arn:aws:lambda:us-east-1:111222333444:function:SSMTestLAMBDA \
```

```
    --service-role-arn arn:aws:iam::111222333444:role/SSM \
    --task-type LAMBDA \
    --task-invocation-parameters '{"Lambda":{"Payload":"{\"InstanceId\":
\"{{RESOURCE_ID}}\",\"targetType\":\"{{TARGET_TYPE}}\"}","Qualifier":"$LATEST"}}'
 \
    --priority 0 \
    --max-concurrency 10 \
    --max-errors 5 \
    --name "Lambda_Example" \
    --description "My Lambda Example"
```

Sortie :

```
{
    "WindowTaskId":"22244444-5555-6666-7777-88888888"
}
```

Pour plus d'informations, consultez la section Enregistrer une tâche avec la fenêtre de maintenance (AWS CLI) dans le guide de l'utilisateur de AWS Systems Manager.

Exemple 3 : pour enregistrer une tâche Run Command avec une fenêtre de maintenance

L'`register-task-with-maintenance-window`exemple suivant enregistre une tâche Run Command avec une fenêtre de maintenance ciblée sur une instance.

```
aws ssm register-task-with-maintenance-window \
    --window-id "mw-082dcd7649dee04e4" \
    --targets "Key=InstanceIds,Values=i-12344d305eEXAMPLE" \
    --service-role-arn "arn:aws:iam::111222333444:role/SSM" \
    --task-type "RUN_COMMAND" \
    --name "SSMInstallPowerShellModule" \
    --task-arn "AWS-InstallPowerShellModule" \
    --task-invocation-parameters "{\"RunCommand\":{\"Comment\":\"\",
\"OutputS3BucketName\":\"runcommandlogs\",\"Parameters\":{\"commands\":[\"Get-
Module -ListAvailable\"],\"executionTimeout\":[\"3600\"],\"source\":[\"https:\/
\/gallery.technet.microsoft.com\/EZOut-33ae0fb7\/file\/110351\/1\/EZOut.zip\"],
\"workingDirectory\":[\"\\\\\"]},\"TimeoutSeconds\":600}}" \
    --max-concurrency 1 \
    --max-errors 1 \
    --priority 10
```

Sortie :

```
{
    "WindowTaskId":"33344444-5555-6666-7777-88888888"
}
```

Pour plus d'informations, consultez la section Enregistrer une tâche avec la fenêtre de maintenance (AWS CLI) dans le guide de l'utilisateur de AWS Systems Manager.

Exemple 4 : Pour enregistrer une tâche Step Functions avec une fenêtre de maintenance

L'`register-task-with-maintenance-window`exemple suivant enregistre une tâche Step Functions avec une fenêtre de maintenance ciblée sur une instance.

```
aws ssm register-task-with-maintenance-window \
    --window-id "mw-1234d787d6EXAMPLE" \
    --targets Key=WindowTargetIds,Values=12347414-69c3-49f8-95b8-ed2dcEXAMPLE \
    --task-arn arn:aws:states:us-
east-1:111222333444:stateMachine:SSMTestStateMachine \
    --service-role-arn arn:aws:iam::111222333444:role/MaintenanceWindows \
    --task-type STEP_FUNCTIONS \
    --task-invocation-parameters '{"StepFunctions":{"Input":"{\"InstanceId\":
\"{{RESOURCE_ID}}\"}"}}' \
    --priority 0 \
    --max-concurrency 10 \
    --max-errors 5 \
    --name "Step_Functions_Example" \
    --description "My Step Functions Example"
```

Sortie :

```
{
    "WindowTaskId":"44444444-5555-6666-7777-88888888"
}
```

Pour plus d'informations, consultez la section Enregistrer une tâche avec la fenêtre de maintenance (AWS CLI) dans le guide de l'utilisateur de AWS Systems Manager.

Exemple 5 : Pour enregistrer une tâche à l'aide d'un ID cible Windows de maintenance

L'`register-task-with-maintenance-window`exemple suivant enregistre une tâche à l'aide d'un ID cible de fenêtre de maintenance. L'ID cible de la fenêtre de maintenance figurait dans le résultat de la `aws ssm register-target-with-maintenance-`

`window` commande. Vous pouvez également le récupérer à partir de la sortie de la `aws ssm describe-maintenance-window-targets` commande.

```
aws ssm register-task-with-maintenance-window \
    --targets "Key=WindowTargetIds,Values=350d44e6-28cc-44e2-951f-4b2c9EXAMPLE" \
    --task-arn "AWS-RunShellScript" \
    --service-role-arn "arn:aws:iam::111222333444:role/MaintenanceWindowsRole" \
    --window-id "mw-ab12cd34eEXAMPLE" \
    --task-type "RUN_COMMAND" \
    --task-parameters  "{\"commands\":{\"Values\":[\"df\"]}}" \
    --max-concurrency 1 \
    --max-errors 1 \
    --priority 10
```

Sortie :

```
{
    "WindowTaskId":"33344444-5555-6666-7777-88888888"
}
```

Pour plus d'informations, consultez la section Enregistrer une tâche avec la fenêtre de maintenance (AWS CLI) dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, voir RegisterTaskWithMaintenanceWindow in AWS CLI Command Reference.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple enregistre une tâche avec une fenêtre de maintenance à l'aide d'un ID d'instance. Le résultat est l'ID de tâche.

```
$parameters = @{}
$parameterValues = New-Object
 Amazon.SimpleSystemsManagement.Model.MaintenanceWindowTaskParameterValueExpression
$parameterValues.Values = @("Install")
$parameters.Add("Operation", $parameterValues)

Register-SSMTaskWithMaintenanceWindow -WindowId "mw-03a342e62c96d31b0"
 -ServiceRoleArn "arn:aws:iam::123456789012:role/MaintenanceWindowsRole"
```

```
 -MaxConcurrency 1 -MaxError 1 -TaskArn "AWS-RunShellScript" -Target
 @{ Key="InstanceIds";Values="i-0000293ffd8c57862" } -TaskType "RUN_COMMAND" -
Priority 10 -TaskParameter $parameters
```

Sortie :

```
f34a2c47-ddfd-4c85-a88d-72366b69af1b
```

Exemple 2 : Cet exemple enregistre une tâche avec une fenêtre de maintenance à l'aide d'un ID cible. Le résultat est l'ID de tâche.

```
$parameters = @{}
$parameterValues = New-Object
 Amazon.SimpleSystemsManagement.Model.MaintenanceWindowTaskParameterValueExpression
$parameterValues.Values = @("Install")
$parameters.Add("Operation", $parameterValues)

register-ssmtaskwithmaintenancewindow -WindowId "mw-03a342e62c96d31b0"
 -ServiceRoleArn "arn:aws:iam::123456789012:role/MaintenanceWindowsRole"
 -MaxConcurrency 1 -MaxError 1 -TaskArn "AWS-RunShellScript" -Target
 @{ Key="WindowTargetIds";Values="350d44e6-28cc-44e2-951f-4b2c985838f6" } -
TaskType "RUN_COMMAND" -Priority 10 -TaskParameter $parameters
```

Sortie :

```
f34a2c47-ddfd-4c85-a88d-72366b69af1b
```

Exemple 3 : Cet exemple crée un objet de paramètre pour le document de commande d'exécution **AWS-RunPowerShellScript** et crée une tâche avec une fenêtre de maintenance donnée en utilisant l'ID cible. Le résultat renvoyé est l'ID de tâche.

```
$parameters =
 [Collections.Generic.Dictionary[String,Collections.Generic.List[String]]]::new()
$parameters.Add("commands",@("ipconfig","dir env:\computername"))
$parameters.Add("executionTimeout",@(3600))

$props = @{
    WindowId = "mw-0123e4cce56ff78ae"
    ServiceRoleArn = "arn:aws:iam::123456789012:role/MaintenanceWindowsRole"
    MaxConcurrency = 1
```

```
    MaxError = 1
    TaskType = "RUN_COMMAND"
    TaskArn = "AWS-RunPowerShellScript"
    Target =
 @{Key="WindowTargetIds";Values="fe1234ea-56d7-890b-12f3-456b789bee0f"}
    Priority = 1
    RunCommand_Parameter = $parameters
    Name = "set-via-cmdlet"
}

Register-SSMTaskWithMaintenanceWindow @props
```

Sortie :

```
f1e2ef34-5678-12e3-456a-12334c5c6cbe
```

Exemple 4 : Cet exemple enregistre une tâche d'automatisation de AWS Systems Manager à l'aide d'un document nommé**`Create-Snapshots`**.

```
$automationParameters = @{}
$automationParameters.Add( "instanceId", @("{{ TARGET_ID }}") )
$automationParameters.Add( "AutomationAssumeRole",
 @("{arn:aws:iam::111111111111:role/AutomationRole}") )
$automationParameters.Add( "SnapshotTimeout", @("PT20M") )
Register-SSMTaskWithMaintenanceWindow -WindowId mw-123EXAMPLE456`
    -ServiceRoleArn "arn:aws:iam::123456789012:role/MW-Role"`
    -MaxConcurrency 1 -MaxError 1 -TaskArn "CreateVolumeSnapshots"`
    -Target @{ Key="WindowTargetIds";Values="4b5acdf4-946c-4355-
bd68-4329a43a5fd1" }`
    -TaskType "AUTOMATION"`
    -Priority 4`
    -Automation_DocumentVersion '$DEFAULT' -Automation_Parameter
 $automationParameters -Name "Create-Snapshots"
```

- Pour plus de détails sur l'API, consultez RegisterTaskWithMaintenancela section Window in AWS Tools for PowerShell Cmdlet Reference.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **RemoveTagsFromResource** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliserRemoveTagsFromResource.

CLI

AWS CLI

Pour supprimer une balise d'une ligne de base de correctifs

L'remove-tags-from-resourceexemple suivant supprime les balises d'une ligne de base de correctif.

```
aws ssm remove-tags-from-resource \
    --resource-type "PatchBaseline" \
    --resource-id "pb-0123456789abcdef0" \
    --tag-keys "Region"
```

Cette commande ne produit aucun résultat.

Pour plus d'informations, consultez la section AWS Ressources de balisage dans le manuel de référence AWS général.

- Pour plus de détails sur l'API, reportez-vous RemoveTagsFromResourceà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple supprime une balise d'une fenêtre de maintenance. Il n'y a aucune sortie si la commande aboutit.

```
Remove-SSMResourceTag -ResourceId "mw-03eb9db42890fb82d" -ResourceType
 "MaintenanceWindow" -TagKey "Production"
```

- Pour plus de détails sur l'API, reportez-vous RemoveTagsFromResourceà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **SendCommand** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliserSendCommand.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- Commencez avec Systems Manager

CLI

AWS CLI

Exemple 1 : pour exécuter une commande sur une ou plusieurs instances distantes

L'send-commandexemple suivant exécute une echo commande sur une instance cible.

```
aws ssm send-command \
    --document-name "AWS-RunShellScript" \
    --parameters 'commands=["echo HelloWorld"]' \
    --targets "Key=instanceids,Values=i-1234567890abcdef0" \
    --comment "echo HelloWorld"
```

Sortie :

```
{
    "Command": {
        "CommandId": "92853adf-ba41-4cd6-9a88-142d1EXAMPLE",
        "DocumentName": "AWS-RunShellScript",
        "DocumentVersion": "",
        "Comment": "echo HelloWorld",
        "ExpiresAfter": 1550181014.717,
        "Parameters": {
            "commands": [
                "echo HelloWorld"
            ]
```

```
        },
        "InstanceIds": [
            "i-0f00f008a2dcbefe2"
        ],
        "Targets": [],
        "RequestedDateTime": 1550173814.717,
        "Status": "Pending",
        "StatusDetails": "Pending",
        "OutputS3BucketName": "",
        "OutputS3KeyPrefix": "",
        "MaxConcurrency": "50",
        "MaxErrors": "0",
        "TargetCount": 1,
        "CompletedCount": 0,
        "ErrorCount": 0,
        "DeliveryTimedOutCount": 0,
        "ServiceRole": "",
        "NotificationConfig": {
            "NotificationArn": "",
            "NotificationEvents": [],
            "NotificationType": ""
        },
        "CloudWatchOutputConfig": {
            "CloudWatchLogGroupName": "",
            "CloudWatchOutputEnabled": false
        }
    }
}
```

Pour plus d'informations, consultez la section Exécution de commandes à l'aide de Systems Manager Run Command dans le Guide de l'utilisateur de AWS Systems Manager.

Exemple 2 : Pour obtenir des informations IP sur une instance

L'`send-command`exemple suivant récupère les informations IP relatives à une instance.

```
aws ssm send-command \
    --instance-ids "i-1234567890abcdef0" \
    --document-name "AWS-RunShellScript" \
    --comment "IP config" \
    --parameters "commands=ifconfig"
```

Voir l'exemple 1 pour un exemple de sortie.

Pour plus d'informations, consultez la section Exécution de commandes à l'aide de Systems Manager Run Command dans le Guide de l'utilisateur de AWS Systems Manager.

Exemple 3 : pour exécuter une commande sur des instances comportant des balises spécifiques

L'`send-command`exemple suivant exécute une commande sur des instances dont le tag est « ENV » et la valeur « Dev ».

```
aws ssm send-command \
    --targets "Key=tag:ENV,Values=Dev" \
    --document-name "AWS-RunShellScript" \
    --parameters "commands=ifconfig"
```

Voir l'exemple 1 pour un exemple de sortie.

Pour plus d'informations, consultez la section Exécution de commandes à l'aide de Systems Manager Run Command dans le Guide de l'utilisateur de AWS Systems Manager.

Exemple 4 : pour exécuter une commande qui envoie des notifications SNS

L'`send-command`exemple suivant exécute une commande qui envoie des notifications SNS pour tous les événements de notification et le type de `Command` notification.

```
aws ssm send-command \
    --instance-ids "i-1234567890abcdef0" \
    --document-name "AWS-RunShellScript" \
    --comment "IP config" \
    --parameters "commands=ifconfig" \
    --service-role-arn "arn:aws:iam::123456789012:role/SNS_Role" \
    --notification-config "NotificationArn=arn:aws:sns:us-
east-1:123456789012:SNSTopicName,NotificationEvents=All,NotificationType=Command"
```

Voir l'exemple 1 pour un exemple de sortie.

Pour plus d'informations, consultez la section Exécution de commandes à l'aide de Systems Manager Run Command dans le Guide de l'utilisateur de AWS Systems Manager.

Exemple 5 : Pour exécuter une commande qui renvoie vers S3 et CloudWatch

L'`send-command`exemple suivant exécute une commande qui affiche les détails de la commande dans un compartiment S3 et dans un groupe de CloudWatch journaux Logs.

```
aws ssm send-command \
    --instance-ids "i-1234567890abcdef0" \
    --document-name "AWS-RunShellScript" \
    --comment "IP config" \
    --parameters "commands=ifconfig" \
    --output-s3-bucket-name "s3-bucket-name" \
    --output-s3-key-prefix "runcommand" \
    --cloud-watch-output-config
 "CloudWatchOutputEnabled=true,CloudWatchLogGroupName=CWLGroupName"
```

Voir l'exemple 1 pour un exemple de sortie.

Pour plus d'informations, consultez la section Exécution de commandes à l'aide de Systems Manager Run Command dans le Guide de l'utilisateur de AWS Systems Manager.

Exemple 6 : pour exécuter des commandes sur plusieurs instances avec des balises différentes

L'`send-command`exemple suivant exécute une commande sur des instances comportant deux clés de balise et deux valeurs différentes.

```
aws ssm send-command \
    --document-name "AWS-RunPowerShellScript" \
    --parameters commands=["echo helloWorld"] \
    --targets Key=tag:Env,Values=Dev Key=tag:Role,Values=WebServers
```

Voir l'exemple 1 pour un exemple de sortie.

Pour plus d'informations, consultez la section Exécution de commandes à l'aide de Systems Manager Run Command dans le Guide de l'utilisateur de AWS Systems Manager.

Exemple 7 : Pour cibler plusieurs instances avec la même clé de balise

L'`send-command`exemple suivant exécute une commande sur des instances qui ont la même clé de balise mais avec des valeurs différentes.

```
aws ssm send-command \
    --document-name "AWS-RunPowerShellScript" \
    --parameters commands=["echo helloWorld"] \
    --targets Key=tag:Env,Values=Dev,Test
```

Voir l'exemple 1 pour un exemple de sortie.

Pour plus d'informations, consultez la section Exécution de commandes à l'aide de Systems Manager Run Command dans le Guide de l'utilisateur de AWS Systems Manager.

Exemple 8 : Pour exécuter une commande utilisant un document partagé

L'`send-command`exemple suivant exécute un document partagé sur une instance cible.

```
aws ssm send-command \
    --document-name "arn:aws:ssm:us-east-1:123456789012:document/ExampleDocument"
 \
    --targets "Key=instanceids,Values=i-1234567890abcdef0"
```

Voir l'exemple 1 pour un exemple de sortie.

Pour plus d'informations, consultez la section Utilisation de documents SSM partagés dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous SendCommandà la section Référence des AWS CLI commandes.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
   // Sends a SSM command to a managed node.
   public static String sendSSMCommand(SsmClient ssmClient, String documentName,
String instanceId) throws InterruptedException {
       // Before we use Document to send a command - make sure it is active.
       boolean isDocumentActive = false;
       DescribeDocumentRequest request = DescribeDocumentRequest.builder()
           .name(documentName)
           .build();

       while (!isDocumentActive) {
           DescribeDocumentResponse response =
ssmClient.describeDocument(request);
```

```
            String documentStatus = response.document().statusAsString();
            if (documentStatus.equals("Active")) {
                System.out.println("The Systems Manager document is active and
ready to use.");
                isDocumentActive = true;
            } else {
                System.out.println("The Systems Manager document is not active.
Status: " + documentStatus);
                try {
                    // Add a delay to avoid making too many requests.
                    Thread.sleep(5000); // Wait for 5 seconds before checking
again
                } catch (InterruptedException e) {
                    e.printStackTrace();
                }
            }
        }

        // Create the SendCommandRequest.
        SendCommandRequest commandRequest = SendCommandRequest.builder()
            .documentName(documentName)
            .instanceIds(instanceId)
            .build();

        // Send the command.
        SendCommandResponse commandResponse =
ssmClient.sendCommand(commandRequest);
        String commandId = commandResponse.command().commandId();
        System.out.println("The command Id is " + commandId);

        // Wait for the command execution to complete.
        GetCommandInvocationRequest invocationRequest =
GetCommandInvocationRequest.builder()
            .commandId(commandId)
            .instanceId(instanceId)
            .build();

        System.out.println("Wait 5 secs");
        TimeUnit.SECONDS.sleep(5);

        // Retrieve the command execution details.
        GetCommandInvocationResponse commandInvocationResponse =
ssmClient.getCommandInvocation(invocationRequest);
```

```
        // Check the status of the command execution.
        CommandInvocationStatus status = commandInvocationResponse.status();
        if (status == CommandInvocationStatus.SUCCESS) {
            System.out.println("Command execution successful.");
        } else {
            System.out.println("Command execution failed. Status: " + status);
        }
        return commandId;
    }
```

- Pour plus de détails sur l'API, reportez-vous SendCommandà la section Référence des AWS SDK for Java 2.x API.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple exécute une commande echo sur une instance cible.

```
Send-SSMCommand -DocumentName "AWS-RunPowerShellScript" -Parameter @{commands =
 "echo helloWorld"} -Target @{Key="instanceids";Values=@("i-0cb2b964d3e14fd9f")}
```

Sortie :

```
CommandId          : d8d190fc-32c1-4d65-a0df-ff5ff3965524
Comment            :
CompletedCount     : 0
DocumentName       : AWS-RunPowerShellScript
ErrorCount         : 0
ExpiresAfter       : 3/7/2017 10:48:37 PM
InstanceIds        : {}
MaxConcurrency     : 50
MaxErrors          : 0
NotificationConfig : Amazon.SimpleSystemsManagement.Model.NotificationConfig
OutputS3BucketName :
OutputS3KeyPrefix  :
OutputS3Region     :
Parameters         : {[commands,
 Amazon.Runtime.Internal.Util.AlwaysSendList`1[System.String]]}
RequestedDateTime  : 3/7/2017 9:48:37 PM
ServiceRole        :
```

```
Status              : Pending
StatusDetails       : Pending
TargetCount         : 0
Targets             : {instanceids}
```

Exemple 2 : Cet exemple montre comment exécuter une commande qui accepte des paramètres imbriqués.

```
Send-SSMCommand -DocumentName "AWS-RunRemoteScript" -Parameter
 @{ sourceType="GitHub";sourceInfo='{"owner": "me","repository": "amazon-
ssm","path": "Examples/Install-Win32OpenSSH"}'; "commandLine"=".\Install-
Win32OpenSSH.ps1"} -InstanceId i-0cb2b964d3e14fd9f
```

- Pour plus de détails sur l'API, reportez-vous SendCommandà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`StartAutomationExecution`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`StartAutomationExecution`.

CLI

AWS CLI

Exemple 1 : pour exécuter un document d'automatisation

L'`start-automation-execution`exemple suivant exécute un document d'automatisation.

```
aws ssm start-automation-execution \
    --document-name "AWS-UpdateLinuxAmi" \
    --parameters "AutomationAssumeRole=arn:aws:iam::123456789012:role/
SSMAutomationRole,SourceAmiId=ami-EXAMPLE,IamInstanceProfileName=EC2InstanceRole"
```

Sortie :

```
{
  "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0a1b2EXAMPLE"
```

```
}
```

Pour plus d'informations, consultez la section Exécution manuelle d'un flux de travail d'automatisation dans le guide de l'utilisateur de AWS Systems Manager.

Exemple 2 : pour exécuter un document d'automatisation partagé

L'`start-automation-execution`exemple suivant exécute un document d'automatisation partagé.

```
aws ssm start-automation-execution \
    --document-name "arn:aws:ssm:us-east-1:123456789012:document/ExampleDocument"
```

Sortie :

```
{
  "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0a1b2EXAMPLE"
}
```

Pour plus d'informations, consultez la section Utilisation de documents SSM partagés dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, voir StartAutomationExécution dans AWS CLI la référence des commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple exécute un document spécifiant un rôle d'automatisation, un ID de source AMI et un rôle d'instance Amazon EC2.

```
Start-SSMAutomationExecution -DocumentName AWS-UpdateLinuxAmi -
Parameter @{'AutomationAssumeRole'='arn:aws:iam::123456789012:role/
SSMAutomationRole';'SourceAmiId'='ami-
f173cc91';'InstanceIamRole'='EC2InstanceRole'}
```

Sortie :

```
3a532a4f-0382-11e7-9df7-6f11185f6dd1
```

- Pour plus de détails sur l'API, consultez la section StartAutomationExecution in AWS Tools for PowerShell Cmdlet Reference.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`StopAutomationExecution`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`StopAutomationExecution`.

CLI

AWS CLI

Pour arrêter l'exécution d'une automatisation

L'`stop-automation-execution`exemple suivant arrête un document d'automatisation.

```
aws ssm stop-automation-execution
    --automation-execution-id "4105a4fc-f944-11e6-9d32-0a1b2EXAMPLE"
```

Cette commande ne produit aucun résultat.

Pour plus d'informations, consultez la section Exécution manuelle d'un flux de travail d'automatisation dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, consultez la section StopAutomationExécution dans AWS CLI la référence des commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple arrête une exécution automatique. Il n'y a aucune sortie si la commande aboutit.

```
Stop-SSMAutomationExecution -AutomationExecutionId "4105a4fc-
f944-11e6-9d32-8fb2db27a909"
```

- Pour plus de détails sur l'API, consultez la section StopAutomationExecution in AWS Tools for PowerShell Cmdlet Reference.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`UpdateAssociation`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`UpdateAssociation`.

CLI

AWS CLI

Exemple 1 : pour mettre à jour une association de documents

L'`update-association`exemple suivant met à jour une association avec une nouvelle version de document.

```
aws ssm update-association \
    --association-id "8dfe3659-4309-493a-8755-0123456789ab" \
    --document-version "\$LATEST"
```

Sortie :

```
{
    "AssociationDescription": {
        "Name": "AWS-UpdateSSMAgent",
        "AssociationVersion": "2",
        "Date": 1550508093.293,
        "LastUpdateAssociationDate": 1550508106.596,
        "Overview": {
            "Status": "Pending",
            "DetailedStatus": "Creating"
        },
        "DocumentVersion": "$LATEST",
        "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
        "Targets": [
            {
                "Key": "tag:Name",
                "Values": [
```

```
                    "Linux"
                ]
            }
        ],
        "LastExecutionDate": 1550508094.879,
        "LastSuccessfulExecutionDate": 1550508094.879
    }
}
```

Pour plus d'informations, reportez-vous à la section Modification et création d'une nouvelle version d'une association dans le Guide de l'utilisateur de AWS Systems Manager.

Exemple 2 : pour mettre à jour l'expression de planification d'une association

L'`update-association`exemple suivant met à jour l'expression de planification pour l'association spécifiée.

```
aws ssm update-association \
    --association-id "8dfe3659-4309-493a-8755-0123456789ab" \
    --schedule-expression "cron(0 0 0/4 1/1 * ? *)"
```

Sortie :

```
{
    "AssociationDescription": {
        "Name": "AWS-HelloWorld",
        "AssociationVersion": "2",
        "Date": "2021-02-08T13:54:19.203000-08:00",
        "LastUpdateAssociationDate": "2021-06-29T11:51:07.933000-07:00",
        "Overview": {
            "Status": "Pending",
            "DetailedStatus": "Creating"
        },
        "DocumentVersion": "$DEFAULT",
        "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
        "Targets": [
            {
                "Key": "aws:NoOpAutomationTag",
                "Values": [
                    "AWS-NoOpAutomationTarget-Value"
                ]
            }
        ],
```

```
        "ScheduleExpression": "cron(0 0 0/4 1/1 * ? *)",
        "LastExecutionDate": "2021-06-26T19:00:48.110000-07:00",
        "ApplyOnlyAtCronInterval": false
    }
}
```

Pour plus d'informations, reportez-vous à la section Modification et création d'une nouvelle version d'une association dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous UpdateAssociationà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple met à jour une association avec une nouvelle version de document.

```
Update-SSMAssociation -AssociationId "93285663-92df-44cb-9f26-2292d4ecc439" -
DocumentVersion "1"
```

Sortie :

```
Name                  : AWS-UpdateSSMAgent
InstanceId            :
Date                  : 3/1/2017 6:22:21 PM
Status.Name           :
Status.Date           :
Status.Message        :
Status.AdditionalInfo :
```

- Pour plus de détails sur l'API, consultez la section UpdateAssociationRéférence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`UpdateAssociationStatus`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`UpdateAssociationStatus`.

CLI

AWS CLI

Pour mettre à jour le statut de l'association

L'`update-association-status`exemple suivant met à jour le statut de l'association entre une instance et un document.

```
aws ssm update-association-status \
    --name "AWS-UpdateSSMAgent" \
    --instance-id "i-1234567890abcdef0" \
    --association-status
 "Date=1424421071.939,Name=Pending,Message=temp_status_change,AdditionalInfo=Additional-
Config-Needed"
```

Sortie :

```
{
    "AssociationDescription": {
        "Name": "AWS-UpdateSSMAgent",
        "InstanceId": "i-1234567890abcdef0",
        "AssociationVersion": "1",
        "Date": 1550507529.604,
        "LastUpdateAssociationDate": 1550507806.974,
        "Status": {
            "Date": 1424421071.0,
            "Name": "Pending",
            "Message": "temp_status_change",
            "AdditionalInfo": "Additional-Config-Needed"
        },
        "Overview": {
            "Status": "Success",
            "AssociationStatusAggregatedCount": {
                "Success": 1
            }
        },
        "DocumentVersion": "$DEFAULT",
        "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
        "Targets": [
            {
                "Key": "InstanceIds",
                "Values": [
```

```
                    "i-1234567890abcdef0"
                ]
            }
        ],
        "LastExecutionDate": 1550507808.0,
        "LastSuccessfulExecutionDate": 1550507808.0
    }
}
```

Pour plus d'informations, reportez-vous à la section Utilisation des associations dans Systems Manager dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, consultez la section UpdateAssociationÉtat dans AWS CLI la référence des commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple met à jour le statut d'association de l'association entre une instance et un document de configuration.

```
Update-SSMAssociationStatus -Name "AWS-UpdateSSMAgent" -InstanceId
 "i-0000293ffd8c57862" -AssociationStatus_Date "2015-02-20T08:31:11Z"
 -AssociationStatus_Name "Pending" -AssociationStatus_Message
 "temporary_status_change" -AssociationStatus_AdditionalInfo "Additional-Config-
Needed"
```

Sortie :

```
Name                  : AWS-UpdateSSMAgent
InstanceId            : i-0000293ffd8c57862
Date                  : 2/23/2017 6:55:22 PM
Status.Name           : Pending
Status.Date           : 2/20/2015 8:31:11 AM
Status.Message        : temporary_status_change
Status.AdditionalInfo : Additional-Config-Needed
```

- Pour plus de détails sur l'API, consultez la section UpdateAssociationStatus in AWS Tools for PowerShell Cmdlet Reference.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`UpdateDocument`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`UpdateDocument`.

CLI

AWS CLI

Pour créer une nouvelle version d'un document

L'`update-document`exemple suivant crée une nouvelle version d'un document lorsqu'il est exécuté sur un ordinateur Windows. Le document spécifié par `--document` doit être au format JSON. Notez qu'`file://`il doit être référencé, suivi du chemin du fichier de contenu. En raison du fait qu'`$`au début du `--document-version` paramètre, sous Windows, vous devez entourer la valeur de guillemets doubles. Sous Linux, macOS ou lorsque vous y êtes PowerShell invité, vous devez placer la valeur entre guillemets simples.

Version Windows :

```
aws ssm update-document \
    --name "RunShellScript" \
    --content "file://RunShellScript.json" \
    --document-version "$LATEST"
```

Version Linux/Mac :

```
aws ssm update-document \
    --name "RunShellScript" \
    --content "file://RunShellScript.json" \
    --document-version '$LATEST'
```

Sortie :

```
{
  "DocumentDescription": {
      "Status": "Updating",
      "Hash": "f775e5df4904c6fa46686c4722fae9de1950dace25cd9608ff8d622046b68d9b",
```

```
      "Name": "RunShellScript",
      "Parameters": [
          {
              "Type": "StringList",
              "Name": "commands",
              "Description": "(Required) Specify a shell script or a command to
 run."
          }
      ],
      "DocumentType": "Command",
      "PlatformTypes": [
          "Linux"
      ],
      "DocumentVersion": "2",
      "HashType": "Sha256",
      "CreatedDate": 1487899655.152,
      "Owner": "809632081692",
      "SchemaVersion": "2.0",
      "DefaultVersion": "1",
      "LatestVersion": "2",
      "Description": "Run an updated script"
  }
}
```

- Pour plus de détails sur l'API, reportez-vous UpdateDocumentà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cela crée une nouvelle version d'un document avec le contenu mis à jour du fichier JSON que vous spécifiez. Le document doit être au format JSON. Vous pouvez obtenir la version du document à l'aide de l'applet de commande « Get-SSM DocumentVersion List ».

```
Update-SSMDocument -Name RunShellScript -DocumentVersion "1" -Content (Get-
Content -Raw "c:\temp\RunShellScript.json")
```

Sortie :

```
CreatedDate     : 3/1/2017 2:59:17 AM
DefaultVersion  : 1
```

```
Description     : Run an updated script
DocumentType    : Command
DocumentVersion : 2
Hash            :
 1d5ce820e999ff051eb4841ed887593daf77120fd76cae0d18a53cc42e4e22c1
HashType        : Sha256
LatestVersion   : 2
Name            : RunShellScript
Owner           : 809632081692
Parameters      : {commands}
PlatformTypes   : {Linux}
SchemaVersion   : 2.0
Sha1            :
Status          : Updating
```

- Pour plus de détails sur l'API, reportez-vous UpdateDocumentà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **UpdateDocumentDefaultVersion** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`UpdateDocumentDefaultVersion`.

CLI

AWS CLI

Pour mettre à jour la version par défaut d'un document

L'`update-document-default-version`exemple suivant met à jour la version par défaut d'un document Systems Manager.

```
aws ssm update-document-default-version \
    --name "Example" \
    --document-version "2"
```

Sortie :

```
{
    "Description": {
        "Name": "Example",
        "DefaultVersion": "2"
    }
}
```

Pour plus d'informations, consultez la section Rédaction du contenu d'un document SSM dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous UpdateDocumentDefaultVersionà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Ceci met à jour la version par défaut d'un document. Vous pouvez obtenir les versions de document disponibles à l'aide de l'applet de commande « Get-SSM DocumentVersion List ».

```
Update-SSMDocumentDefaultVersion -Name "RunShellScript" -DocumentVersion "2"
```

Sortie :

```
DefaultVersion Name
-------------- ----
2              RunShellScript
```

- Pour plus de détails sur l'API, reportez-vous UpdateDocumentDefaultVersionà la section Référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`UpdateMaintenanceWindow`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`UpdateMaintenanceWindow`.

CLI

AWS CLI

Exemple 1 : pour mettre à jour une fenêtre de maintenance

L'`update-maintenance-window`exemple suivant met à jour le nom d'une fenêtre de maintenance.

```
aws ssm update-maintenance-window \
    --window-id "mw-1a2b3c4d5e6f7g8h9" \
    --name "My-Renamed-MW"
```

Sortie :

```
{
    "Cutoff": 1,
    "Name": "My-Renamed-MW",
    "Schedule": "cron(0 16 ? * TUE *)",
    "Enabled": true,
    "AllowUnassociatedTargets": true,
    "WindowId": "mw-1a2b3c4d5e6f7g8h9",
    "Duration": 4
}
```

Exemple 2 : pour désactiver une fenêtre de maintenance

L'`update-maintenance-window`exemple suivant désactive une fenêtre de maintenance.

```
aws ssm update-maintenance-window \
    --window-id "mw-1a2b3c4d5e6f7g8h9" \
    --no-enabled
```

Exemple 3 : pour activer une fenêtre de maintenance

L'`update-maintenance-window`exemple suivant active une fenêtre de maintenance.

```
aws ssm update-maintenance-window \
    --window-id "mw-1a2b3c4d5e6f7g8h9" \
    --enabled
```

Pour plus d'informations, voir Mettre à jour une fenêtre de maintenance (AWS CLI) dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, voir UpdateMaintenanceWindow in AWS CLI Command Reference.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
   // Update the maintenance window schedule
   public static void updateSSMMaintenanceWindow(SsmClient ssmClient, String id,
String name) {
       try {
           UpdateMaintenanceWindowRequest updateRequest =
UpdateMaintenanceWindowRequest.builder()
                .windowId(id)
                .allowUnassociatedTargets(true)
                .duration(24)
                .enabled(true)
                .name(name)
                .schedule("cron(0 0 ? * MON *)")
                .build();

           ssmClient.updateMaintenanceWindow(updateRequest);
           System.out.println("The Systems Manager maintenance window was
successfully updated.");

       } catch (SsmException e) {
           System.err.println(e.getMessage());
           System.exit(1);
       }
   }
```

- Pour plus de détails sur l'API, voir UpdateMaintenanceWindow in AWS SDK for Java 2.x API Reference.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple met à jour le nom d'une fenêtre de maintenance.

```
Update-SSMMaintenanceWindow -WindowId "mw-03eb9db42890fb82d" -Name "My-Renamed-
MW"
```

Sortie :

```
AllowUnassociatedTargets : False
Cutoff                   : 1
Duration                 : 2
Enabled                  : True
Name                     : My-Renamed-MW
Schedule                 : cron(0 */30 * * * ? *)
WindowId                 : mw-03eb9db42890fb82d
```

Exemple 2 : Cet exemple active une fenêtre de maintenance.

```
Update-SSMMaintenanceWindow -WindowId "mw-03eb9db42890fb82d" -Enabled $true
```

Sortie :

```
AllowUnassociatedTargets : False
Cutoff                   : 1
Duration                 : 2
Enabled                  : True
Name                     : My-Renamed-MW
Schedule                 : cron(0 */30 * * * ? *)
WindowId                 : mw-03eb9db42890fb82d
```

Exemple 3 : Cet exemple désactive une fenêtre de maintenance.

```
Update-SSMMaintenanceWindow -WindowId "mw-03eb9db42890fb82d" -Enabled $false
```

Sortie :

```
AllowUnassociatedTargets : False
Cutoff                   : 1
Duration                 : 2
Enabled                  : False
Name                     : My-Renamed-MW
Schedule                 : cron(0 */30 * * * ? *)
WindowId                 : mw-03eb9db42890fb82d
```

- Pour plus de détails sur l'API, consultez UpdateMaintenancela section Window in AWS Tools for PowerShell Cmdlet Reference.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`UpdateManagedInstanceRole`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`UpdateManagedInstanceRole`.

CLI

AWS CLI

Pour mettre à jour le rôle IAM d'une instance gérée

L'`update-managed-instance-role`exemple suivant met à jour le profil d'instance IAM d'une instance gérée.

```
aws ssm update-managed-instance-role \
    --instance-id "mi-08ab247cdfEXAMPLE" \
    --iam-role "ExampleRole"
```

Cette commande ne produit aucun résultat.

Pour plus d'informations, reportez-vous à Etape 4 : Création d'un profil d'instance IAM pour Systems Manager dans le Guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, voir UpdateManagedInstanceRolela section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple met à jour le rôle d'une instance gérée. Il n'y a aucune sortie si la commande aboutit.

```
Update-SSMManagedInstanceRole -InstanceId "mi-08ab247cdf1046573" -IamRole
 "AutomationRole"
```

- Pour plus de détails sur l'API, consultez la section UpdateManagedInstanceRoleRéférence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`UpdateOpsItem`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`UpdateOpsItem`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- Commencez avec Systems Manager

CLI

AWS CLI

Pour mettre à jour un OpsItem

L'`update-ops-item`exemple suivant met à jour la description, la priorité et la catégorie d'un OpsItem. En outre, la commande spécifie une rubrique SNS dans laquelle les notifications sont envoyées lorsqu'elle OpsItem est modifiée ou modifiée.

```
aws ssm update-ops-item \
    --ops-item-id "oi-287b5EXAMPLE" \
```

```
    --description "Primary OpsItem for failover event 2020-01-01-fh398yf" \
    --priority 2 \
    --category "Security" \
    --notifications "Arn=arn:aws:sns:us-east-2:111222333444:my-us-east-2-topic"
```

Sortie :

```
This command produces no output.
```

Pour plus d'informations, reportez-vous à la section Travailler avec OpsItems dans le guide de l'utilisateur de AWS Systems Manager.

- Pour plus de détails sur l'API, reportez-vous à la section UpdateOpsElément du manuel de référence des AWS CLI commandes.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
public static void resolveOpsItem(SsmClient ssmClient, String opsID) {
    try {
        UpdateOpsItemRequest opsItemRequest = UpdateOpsItemRequest.builder()
            .opsItemId(opsID)
            .status(OpsItemStatus.RESOLVED)
            .build();

        ssmClient.updateOpsItem(opsItemRequest);

    } catch (SsmException e) {
        System.err.println(e.getMessage());
        System.exit(1);
    }
}
```

- Pour plus de détails sur l'API, reportez-vous à la section UpdateOpsElément du manuel de référence des AWS SDK for Java 2.x API.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`UpdatePatchBaseline`** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser`UpdatePatchBaseline`.

CLI

AWS CLI

Exemple 1 : pour mettre à jour une ligne de base de correctifs

L'`update-patch-baseline`exemple suivant ajoute les deux correctifs spécifiés comme rejetés et un correctif approuvé à la ligne de base de correctifs spécifiée.

```
aws ssm update-patch-baseline \
        --baseline-id "pb-0123456789abcdef0" \
        --rejected-patches "KB2032276" "MS10-048" \
        --approved-patches "KB2124261"
```

Sortie :

```
{
    "BaselineId": "pb-0123456789abcdef0",
    "Name": "WindowsPatching",
    "OperatingSystem": "WINDOWS",
    "GlobalFilters": {
        "PatchFilters": []
    },
    "ApprovalRules": {
        "PatchRules": [
            {
                "PatchFilterGroup": {
                    "PatchFilters": [
                        {
                            "Key": "PRODUCT",
                            "Values": [
```

```
                                "WindowsServer2016"
                            ]
                        }
                    ]
                },
                "ComplianceLevel": "CRITICAL",
                "ApproveAfterDays": 0,
                "EnableNonSecurity": false
            }
        ]
    },
    "ApprovedPatches": [
        "KB2124261"
    ],
    "ApprovedPatchesComplianceLevel": "UNSPECIFIED",
    "ApprovedPatchesEnableNonSecurity": false,
    "RejectedPatches": [
        "KB2032276",
        "MS10-048"
    ],
    "RejectedPatchesAction": "ALLOW_AS_DEPENDENCY",
    "CreatedDate": 1550244180.465,
    "ModifiedDate": 1550244180.465,
    "Description": "Patches for Windows Servers",
    "Sources": []
}
```

Exemple 2 : pour renommer une ligne de base de correctif

L'`update-patch-baseline`exemple suivant renomme la ligne de base de correctif spécifiée.

```
aws ssm update-patch-baseline \
    --baseline-id "pb-0713accee01234567" \
    --name "Windows-Server-2012-R2-Important-and-Critical-Security-Updates"
```

Pour plus d'informations, consultez la section Mettre à jour ou supprimer une référence de patch`__ dans le guide de l'utilisateur de Systems AWS Manager. < https://docs.aws.amazon.com/systems-manager/latest/userguide/patch-baseline-update-or-delete.html>

- Pour plus de détails sur l'API, voir UpdatePatchBaseline dans AWS CLI Command Reference.

PowerShell

Outils pour PowerShell

Exemple 1 : Cet exemple ajoute deux correctifs rejetés et un correctif approuvé à une ligne de base de correctifs existante.

```
Update-SSMPatchBaseline -BaselineId "pb-03da896ca3b68b639" -RejectedPatch
 "KB2032276","MS10-048" -ApprovedPatch "KB2124261"
```

Sortie :

```
ApprovalRules   : Amazon.SimpleSystemsManagement.Model.PatchRuleGroup
ApprovedPatches : {KB2124261}
BaselineId      : pb-03da896ca3b68b639
CreatedDate     : 3/3/2017 5:02:19 PM
Description     : Baseline containing all updates approved for production systems
GlobalFilters   : Amazon.SimpleSystemsManagement.Model.PatchFilterGroup
ModifiedDate    : 3/3/2017 5:22:10 PM
Name            : Production-Baseline
RejectedPatches : {KB2032276, MS10-048}
```

- Pour plus de détails sur l'API, consultez la section UpdatePatchBaseline dans la référence des AWS Tools for PowerShell applets de commande.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Scénarios pour Systems Manager utilisant des AWS SDK

Les exemples de code suivants vous montrent comment implémenter des scénarios courants dans Systems Manager avec des AWS SDK. Ces scénarios vous montrent comment accomplir des tâches spécifiques en appelant plusieurs fonctions dans Systems Manager. Chaque scénario inclut un lien vers GitHub, où vous pouvez trouver des instructions sur la façon de configurer et d'exécuter le code.

Exemples

- Commencez à utiliser Systems Manager à l'aide d'un AWS SDK

Commencez à utiliser Systems Manager à l'aide d'un AWS SDK

L'exemple de code suivant montre comment utiliser les fenêtres de maintenance de Systems Manager, les documents et OpsItems.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.ssm.SsmClient;
import software.amazon.awssdk.services.ssm.model.CommandInvocation;
import software.amazon.awssdk.services.ssm.model.CommandInvocationStatus;
import software.amazon.awssdk.services.ssm.model.CreateDocumentRequest;
import software.amazon.awssdk.services.ssm.model.CreateDocumentResponse;
import software.amazon.awssdk.services.ssm.model.CreateMaintenanceWindowRequest;
import software.amazon.awssdk.services.ssm.model.CreateMaintenanceWindowResponse;
import software.amazon.awssdk.services.ssm.model.CreateOpsItemRequest;
import software.amazon.awssdk.services.ssm.model.CreateOpsItemResponse;
import software.amazon.awssdk.services.ssm.model.DeleteDocumentRequest;
import software.amazon.awssdk.services.ssm.model.DeleteMaintenanceWindowRequest;
import software.amazon.awssdk.services.ssm.model.DeleteOpsItemRequest;
import software.amazon.awssdk.services.ssm.model.DescribeDocumentRequest;
import software.amazon.awssdk.services.ssm.model.DescribeDocumentResponse;
import
 software.amazon.awssdk.services.ssm.model.DescribeMaintenanceWindowsRequest;
import
 software.amazon.awssdk.services.ssm.model.DescribeMaintenanceWindowsResponse;
import software.amazon.awssdk.services.ssm.model.DescribeOpsItemsRequest;
import software.amazon.awssdk.services.ssm.model.DescribeOpsItemsResponse;
import software.amazon.awssdk.services.ssm.model.DocumentAlreadyExistsException;
import software.amazon.awssdk.services.ssm.model.DocumentType;
import software.amazon.awssdk.services.ssm.model.GetCommandInvocationRequest;
import software.amazon.awssdk.services.ssm.model.GetCommandInvocationResponse;
import software.amazon.awssdk.services.ssm.model.GetOpsItemRequest;
import software.amazon.awssdk.services.ssm.model.GetOpsItemResponse;
```

```
import software.amazon.awssdk.services.ssm.model.ListCommandInvocationsRequest;
import software.amazon.awssdk.services.ssm.model.ListCommandInvocationsResponse;
import software.amazon.awssdk.services.ssm.model.MaintenanceWindowFilter;
import software.amazon.awssdk.services.ssm.model.MaintenanceWindowIdentity;
import software.amazon.awssdk.services.ssm.model.OpsItemDataValue;
import software.amazon.awssdk.services.ssm.model.OpsItemFilter;
import software.amazon.awssdk.services.ssm.model.OpsItemFilterKey;
import software.amazon.awssdk.services.ssm.model.OpsItemFilterOperator;
import software.amazon.awssdk.services.ssm.model.OpsItemStatus;
import software.amazon.awssdk.services.ssm.model.OpsItemSummary;
import software.amazon.awssdk.services.ssm.model.SendCommandRequest;
import software.amazon.awssdk.services.ssm.model.SendCommandResponse;
import software.amazon.awssdk.services.ssm.model.SsmException;
import software.amazon.awssdk.services.ssm.model.UpdateMaintenanceWindowRequest;
import software.amazon.awssdk.services.ssm.model.UpdateOpsItemRequest;
import java.time.ZoneId;
import java.time.format.DateTimeFormatter;
import java.util.HashMap;
import java.util.List;
import java.util.Map;
import java.util.Scanner;
import java.util.concurrent.TimeUnit;

/**
 * Before running this Java V2 code example, set up your development
 * environment, including your credentials.
 *
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/setup.html
 *
 *
 * This Java program performs these tasks:
 * 1. Creates an AWS Systems Manager maintenance window with a default name or a
 user-provided name.
 * 2. Modifies the maintenance window schedule.
 * 3. Creates a Systems Manager document with a default name or a user-provided
 name.
 * 4. Sends a command to a specified EC2 instance using the created Systems
 Manager document and displays the time when the command was invoked.
 * 5. Creates a Systems Manager OpsItem with a predefined title, source,
 category, and severity.
 * 6. Updates and resolves the created OpsItem.
 * 7. Deletes the Systems Manager maintenance window, OpsItem, and document.
```

```
 */

public class SSMScenario {
    public static final String DASHES = new String(new char[80]).replace("\0",
 "-");
    public static void main(String[] args) throws InterruptedException {
        String usage = """
            Usage:
              <instanceId> <title> <source> <category> <severity>

            Where:
                instanceId - The Amazon EC2 Linux/UNIX instance Id that AWS
 Systems Manager uses (ie, i-0149338494ed95f06).
                title - The title of the parameter (default is Disk Space Alert).
                source - The source of the parameter (default is EC2).
                category - The category of the parameter. Valid values are
 'Availability', 'Cost', 'Performance', 'Recovery', 'Security' (default is
 Performance).
                severity - The severity of the parameter. Severity should be a
 number from 1 to 4 (default is 2).
        """;

       if (args.length != 1) {
           System.out.println(usage);
           System.exit(1);
       }

        Scanner scanner = new Scanner(System.in);
        String documentName;
        String windowName;
        String instanceId = args[0];
        String title = "Disk Space Alert" ;
        String source = "EC2" ;
        String category = "Performance" ;
        String severity = "2" ;

        Region region = Region.US_EAST_1;
        SsmClient ssmClient = SsmClient.builder()
            .region(region)
            .build();

        System.out.println(DASHES);
        System.out.println("""
            Welcome to the AWS Systems Manager SDK Getting Started scenario.
```

```
            This program demonstrates how to interact with Systems Manager using
the AWS SDK for Java (v2).
            Systems Manager is the operations hub for your AWS applications and
resources and a secure end-to-end management solution.
            The program's primary functions include creating a maintenance
window, creating a document, sending a command to a document,
            listing documents, listing commands, creating an OpsItem, modifying
an OpsItem, and deleting Systems Manager resources.
            Upon completion of the program, all AWS resources are cleaned up.
            Let's get started...
            Please hit Enter
            """);
        scanner.nextLine();
        System.out.println(DASHES);

        System.out.println("Create a Systems Manager maintenance window.");
        System.out.println("Please enter the maintenance window name (default is
ssm-maintenance-window):");
        String win = scanner.nextLine();
        windowName = win.isEmpty() ? "ssm-maintenance-window" : win;
        String winId = createMaintenanceWindow(ssmClient, windowName);
        System.out.println(DASHES);

        System.out.println("Modify the maintenance window by changing the
schedule");
        System.out.println("Please hit Enter");
        scanner.nextLine();
        updateSSMMaintenanceWindow(ssmClient, winId, windowName);
        System.out.println(DASHES);

        System.out.println("Create a document that defines the actions that
Systems Manager performs on your EC2 instance.");
        System.out.println("Please enter the document name (default is
ssmdocument):");
        String doc = scanner.nextLine();
        documentName = doc.isEmpty() ? "ssmdocument" : doc;
        createSSMDoc(ssmClient, documentName);

        System.out.println("Now we are going to run a command on an EC2 instance
that echoes 'Hello, world!'");
        System.out.println("Please hit Enter");
        scanner.nextLine();
        String commandId = sendSSMCommand(ssmClient, documentName, instanceId);
        System.out.println(DASHES);
```

```
        System.out.println("Lets get the time when the specific command was sent
to the specific managed node");
        System.out.println("Please hit Enter");
        scanner.nextLine();
        displayCommands(ssmClient, commandId);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("""
             Now we will create a  Systems Manager OpsItem.
             An OpsItem is a feature provided by the Systems Manager service.
             It is a type of operational data item that allows you to manage and
track various operational issues,
             events, or tasks within your AWS environment.

             You can create OpsItems to track and manage operational issues as
they arise.
             For example, you could create an OpsItem whenever your application
detects a critical error
             or an anomaly in your infrastructure.
            """);

        System.out.println("Please hit Enter");
        scanner.nextLine();
        String opsItemId = createSSMOpsItem(ssmClient, title, source, category,
severity);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("Now we will update  the OpsItem "+opsItemId);
        System.out.println("Please hit Enter");
        scanner.nextLine();
        String description = "An update to "+opsItemId ;
        updateOpsItem(ssmClient, opsItemId, title, description);
        System.out.println("Now we will get the status of the OpsItem
"+opsItemId);
        System.out.println("Please hit Enter");
        scanner.nextLine();
        describeOpsItems(ssmClient, opsItemId);
        System.out.println("Now we will resolve the OpsItem "+opsItemId);
        System.out.println("Please hit Enter");
        scanner.nextLine();
        resolveOpsItem(ssmClient, opsItemId);
```

```
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("Would you like to delete the Systems Manager
resources? (y/n)");
        String delAns = scanner.nextLine().trim();
        if (delAns.equalsIgnoreCase("y")) {
            System.out.println("You selected to delete the resources.");
            System.out.print("Press Enter to continue...");
            scanner.nextLine();
            deleteOpsItem(ssmClient, opsItemId);
            deleteMaintenanceWindow(ssmClient, winId);
            deleteDoc(ssmClient, documentName);
        } else {
            System.out.println("The Systems Manager resources will not be
deleted");
        }
        System.out.println(DASHES);

        System.out.println("This concludes the Systems Manager SDK Getting
Started scenario.");
        System.out.println(DASHES);
    }

    // Displays the date and time when the specific command was invoked.
    public static void displayCommands(SsmClient ssmClient, String commandId) {
        try {
            ListCommandInvocationsRequest commandInvocationsRequest =
ListCommandInvocationsRequest.builder()
                .commandId(commandId)
                .build();

            ListCommandInvocationsResponse response =
ssmClient.listCommandInvocations(commandInvocationsRequest);
            List<CommandInvocation> commandList = response.commandInvocations();
            DateTimeFormatter formatter = DateTimeFormatter.ofPattern("yyyy-MM-dd
HH:mm:ss").withZone(ZoneId.systemDefault());
            for (CommandInvocation invocation : commandList) {
                System.out.println("The time of the command invocation is " +
formatter.format(invocation.requestedDateTime()));
            }

        } catch (SsmException e) {
            System.err.println(e.getMessage());
```

```
            System.exit(1);
        }
    }

    // Create an SSM OpsItem
    public static String createSSMOpsItem(SsmClient ssmClient, String title,
String source, String category, String severity) {
        try {
            CreateOpsItemRequest opsItemRequest = CreateOpsItemRequest.builder()
                .description("Created by the Systems Manager Java API")
                .title(title)
                .source(source)
                .category(category)
                .severity(severity)
                .build();

            CreateOpsItemResponse itemResponse =
ssmClient.createOpsItem(opsItemRequest);
            return itemResponse.opsItemId();

        } catch (SsmException e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
       return "";
    }

    // Update the AWS SSM OpsItem.
    public static void updateOpsItem(SsmClient ssmClient, String opsItemId,
String title, String description) {
        Map<String, OpsItemDataValue> operationalData = new HashMap<>();
        operationalData.put("key1",
OpsItemDataValue.builder().value("value1").build());
        operationalData.put("key2",
OpsItemDataValue.builder().value("value2").build());

        try {
            UpdateOpsItemRequest request = UpdateOpsItemRequest.builder()
                .opsItemId(opsItemId)
                .title(title)
                .operationalData(operationalData)
                .status(getOpsItem(ssmClient, opsItemId))
                .description(description)
                .build();
```

```
            ssmClient.updateOpsItem(request);

        } catch (SsmException e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }

    public static void resolveOpsItem(SsmClient ssmClient, String opsID) {
        try {
            UpdateOpsItemRequest opsItemRequest = UpdateOpsItemRequest.builder()
                .opsItemId(opsID)
                .status(OpsItemStatus.RESOLVED)
                .build();

            ssmClient.updateOpsItem(opsItemRequest);

        } catch (SsmException e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }

    // Gets a specific OpsItem.
    private static OpsItemStatus getOpsItem(SsmClient ssmClient, String
opsItemId) {
        GetOpsItemRequest itemRequest = GetOpsItemRequest.builder()
            .opsItemId(opsItemId)
            .build();

        try {
            GetOpsItemResponse response = ssmClient.getOpsItem(itemRequest);
            return response.opsItem().status();

        } catch (SsmException e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
        return null;
    }

    // Sends a SSM command to a managed node.
```

```
   public static String sendSSMCommand(SsmClient ssmClient, String documentName,
String instanceId) throws InterruptedException {
       // Before we use Document to send a command - make sure it is active.
       boolean isDocumentActive = false;
       DescribeDocumentRequest request = DescribeDocumentRequest.builder()
           .name(documentName)
           .build();

       while (!isDocumentActive) {
           DescribeDocumentResponse response =
ssmClient.describeDocument(request);
           String documentStatus = response.document().statusAsString();
           if (documentStatus.equals("Active")) {
               System.out.println("The Systems Manager document is active and
ready to use.");
               isDocumentActive = true;
           } else {
               System.out.println("The Systems Manager document is not active.
Status: " + documentStatus);
               try {
                   // Add a delay to avoid making too many requests.
                   Thread.sleep(5000); // Wait for 5 seconds before checking
again
               } catch (InterruptedException e) {
                   e.printStackTrace();
               }
           }
       }

       // Create the SendCommandRequest.
       SendCommandRequest commandRequest = SendCommandRequest.builder()
           .documentName(documentName)
           .instanceIds(instanceId)
           .build();

       // Send the command.
       SendCommandResponse commandResponse =
ssmClient.sendCommand(commandRequest);
       String commandId = commandResponse.command().commandId();
       System.out.println("The command Id is " + commandId);

       // Wait for the command execution to complete.
       GetCommandInvocationRequest invocationRequest =
GetCommandInvocationRequest.builder()
```

```
            .commandId(commandId)
            .instanceId(instanceId)
            .build();

        System.out.println("Wait 5 secs");
        TimeUnit.SECONDS.sleep(5);

        // Retrieve the command execution details.
        GetCommandInvocationResponse commandInvocationResponse =
ssmClient.getCommandInvocation(invocationRequest);

        // Check the status of the command execution.
        CommandInvocationStatus status = commandInvocationResponse.status();
        if (status == CommandInvocationStatus.SUCCESS) {
            System.out.println("Command execution successful.");
        } else {
            System.out.println("Command execution failed. Status: " + status);
        }
        return commandId;
    }

    // Deletes an AWS Systems Manager document.
    public static void deleteDoc(SsmClient ssmClient, String documentName) {
        try {
            DeleteDocumentRequest documentRequest =
DeleteDocumentRequest.builder()
                .name(documentName)
                .build();

            ssmClient.deleteDocument(documentRequest);
            System.out.println("The Systems Manager document was successfully
deleted.");

        } catch (SsmException e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }

    public static void deleteMaintenanceWindow(SsmClient ssmClient, String winId)
{
        try {
            DeleteMaintenanceWindowRequest windowRequest =
DeleteMaintenanceWindowRequest.builder()
```

```
                .windowId(winId)
                .build();

            ssmClient.deleteMaintenanceWindow(windowRequest);
            System.out.println("The maintenance window was successfully
deleted.");

        } catch (SsmException e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }

    // Update the maintenance window schedule
    public static void updateSSMMaintenanceWindow(SsmClient ssmClient, String id,
String name) {
        try {
            UpdateMaintenanceWindowRequest updateRequest =
UpdateMaintenanceWindowRequest.builder()
                .windowId(id)
                .allowUnassociatedTargets(true)
                .duration(24)
                .enabled(true)
                .name(name)
                .schedule("cron(0 0 ? * MON *)")
                .build();

            ssmClient.updateMaintenanceWindow(updateRequest);
            System.out.println("The Systems Manager maintenance window was
successfully updated.");

        } catch (SsmException e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }

    public static String createMaintenanceWindow(SsmClient ssmClient, String
winName) {
        CreateMaintenanceWindowRequest request =
CreateMaintenanceWindowRequest.builder()
            .name(winName)
            .description("This is my maintenance window")
            .allowUnassociatedTargets(true)
```

```
            .duration(2)
            .cutoff(1)
            .schedule("cron(0 10 ? * MON-FRI *)")
            .build();

        try {
            CreateMaintenanceWindowResponse response =
ssmClient.createMaintenanceWindow(request);
            String maintenanceWindowId = response.windowId();
            System.out.println("The maintenance window id is " +
maintenanceWindowId);
            return maintenanceWindowId;

        } catch (DocumentAlreadyExistsException e) {
            System.err.println("The maintenance window already exists. Moving
on.");
        } catch (SsmException e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }

        MaintenanceWindowFilter filter = MaintenanceWindowFilter.builder()
            .key("name")
            .values(winName)
            .build();

        DescribeMaintenanceWindowsRequest winRequest =
DescribeMaintenanceWindowsRequest.builder()
            .filters(filter)
            .build();

        String windowId = "";
        DescribeMaintenanceWindowsResponse response =
ssmClient.describeMaintenanceWindows(winRequest);
        List<MaintenanceWindowIdentity> windows = response.windowIdentities();
        if (!windows.isEmpty()) {
            windowId = windows.get(0).windowId();
            System.out.println("Window ID: " + windowId);
        } else {
            System.out.println("Window not found.");
        }
       return windowId;
    }
```

```
    // Create an AWS SSM document to use in this scenario.
    public static void createSSMDoc(SsmClient ssmClient, String docName) {
        // Create JSON for the content
        String jsonData = """
            {
                "schemaVersion": "2.2",
                "description": "Run a simple shell command",
                "mainSteps": [
                    {
                        "action": "aws:runShellScript",
                        "name": "runEchoCommand",
                        "inputs": {
                          "runCommand": [
                            "echo 'Hello, world!'"
                          ]
                        }
                      }
                    ]
                }
            """;

        try {
            CreateDocumentRequest request = CreateDocumentRequest.builder()
                .content(jsonData)
                .name(docName)
                .documentType(DocumentType.COMMAND)
                .build();

            // Create the document.
            CreateDocumentResponse response = ssmClient.createDocument(request);
            System.out.println("The status of the document is " +
response.documentDescription().status());

        } catch (DocumentAlreadyExistsException e) {
            System.err.println("The document already exists. Moving on." );
        } catch (SsmException e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }

    public static void describeOpsItems(SsmClient ssmClient, String key) {
        try {
            OpsItemFilter filter = OpsItemFilter.builder()
```

```
                .key(OpsItemFilterKey.OPS_ITEM_ID)
                .values(key)
                .operator(OpsItemFilterOperator.EQUAL)
                .build();

            DescribeOpsItemsRequest itemsRequest =
 DescribeOpsItemsRequest.builder()
                .maxResults(10)
                .opsItemFilters(filter)
                .build();

            DescribeOpsItemsResponse itemsResponse =
 ssmClient.describeOpsItems(itemsRequest);
            List<OpsItemSummary> items = itemsResponse.opsItemSummaries();
            for (OpsItemSummary item : items) {
                System.out.println("The item title is " + item.title() +" and the
 status is "+item.status().toString());
            }

        } catch (SsmException e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }

    public static void deleteOpsItem(SsmClient ssmClient, String opsId) {
        try {
            DeleteOpsItemRequest deleteOpsItemRequest =
 DeleteOpsItemRequest.builder()
                .opsItemId(opsId)
                .build();

            ssmClient.deleteOpsItem(deleteOpsItemRequest);
            System.out.println(opsId +" Opsitem was deleted");

        } catch (SsmException e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }
}
```

- Pour plus d'informations sur l'API, consultez les rubriques suivantes dans la référence de l'API AWS SDK for Java 2.x .
  - CommandInvocations
  - CreateDocument
  - CreateMaintenanceFenêtre
  - CreateOpsArticle
  - DeleteMaintenanceFenêtre
  - SendCommand
  - UpdateOpsArticle

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de Systems Manager avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Surveillance AWS Systems Manager

La surveillance joue un rôle important dans le maintien de la fiabilité, de la disponibilité AWS Systems Manager et des performances de vos AWS solutions. Vous devez collecter des données de surveillance provenant de toutes les parties de votre AWS solution afin de pouvoir corriger une défaillance multipoint, le cas échéant. 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 ?
- À quelle fréquence les ressources doivent-elles être surveillées ?
- Quels outils de surveillance 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 de Systems Manager dans votre environnement. Vous devez mesurer les performances de 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 de Systems Manager à leurs données historiques pour vous aider à identifier les modèles de performances normales et les anomalies de performances, et à créer 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. L'automatisation est une capacité de AWS Systems Manager.

Vous pouvez également surveiller l'utilisation de l'UC, les I/O disque et l'utilisation du réseau de vos nœuds gérés. Lorsque les performances ne sont pas conformes à la référence établie, il peut être nécessaire de reconfigurer ou d'optimiser le nœud pour réduire l'utilisation de l'UC, améliorer les I/O disque ou réduire le trafic réseau. Pour plus d'informations sur la surveillance des instances EC2, consultez Surveiller Amazon EC2 dans le guide de l'utilisateur Amazon EC2.

Rubriques

- Envoi des journaux des nœuds vers CloudWatch des journaux unifiés (CloudWatch agent)
- Envoi de journaux SSM Agent à CloudWatch Logs
- Surveillance de vos événements de demande de modification
- Surveillance de vos automatisations
- Surveillance des métriques Run Command avec Amazon CloudWatch
- Journalisation des appels d' AWS Systems Manager API avec AWS CloudTrail
- Journalisation de la sortie d'actions Automation avec CloudWatch Logs
- Configuration d'Amazon CloudWatch Logs pour Run Command
- Surveillance d'événements Systems Manager avec Amazon EventBridge
- Surveillance des changements d'état du Systems Manager à l'aide des notifications Amazon SNS

Outils de surveillance

Le contenu de ce chapitre fournit des informations sur l'utilisation des outils disponibles pour surveiller votre Systems Manager et d'autres AWS ressources. Pour obtenir une liste d'outils plus complète, consultez Journalisation et surveillance dans AWS Systems Manager.

Envoi des journaux des nœuds vers CloudWatch des journaux unifiés (CloudWatch agent)

Vous pouvez configurer et utiliser l' CloudWatch agent Amazon pour collecter des métriques et des journaux à partir de vos nœuds au lieu d'utiliser AWS Systems Manager Agent (SSM Agent) pour ces tâches. L' CloudWatch agent vous permet de collecter plus de métriques sur les instances EC2 que ce qui est disponible avec. SSM Agent En outre, vous pouvez collecter des métriques à partir de serveurs locaux à l'aide de l' CloudWatch agent.

Vous pouvez également enregistrer les paramètres de configuration de l'agent dans le Systems Manager Parameter Store afin de les utiliser avec l' CloudWatch agent. Parameter Storeest une capacité de AWS Systems Manager.

> **Note**
>
> AWS Systems Manager prend en charge la migration depuis SSM Agent l' CloudWatch agent unifié pour collecter des journaux et des métriques uniquement sur les versions 64 bits de Windows. Pour plus d'informations sur la configuration de l' CloudWatch agent unifié

sur d'autres systèmes d'exploitation et pour des informations complètes sur l'utilisation de l' CloudWatch agent, consultez la section Collecte de métriques et de journaux à partir d'instances Amazon EC2 et de serveurs sur site avec l' CloudWatch agent dans le guide de l'utilisateur Amazon CloudWatch .
Vous pouvez utiliser l' CloudWatch agent sur d'autres systèmes d'exploitation pris en charge, mais vous ne pourrez pas utiliser Systems Manager pour effectuer une migration d'outil.

SSM Agent écrit des informations relatives aux exécutions, actions planifiées, erreurs et états d'intégrité dans les fichiers journaux de chaque nœud. La connexion manuelle à un nœud pour afficher les fichiers journaux et résoudre un problème lié à SSM Agent est une opération chronophage. Pour une surveillance plus efficace des nœuds, vous pouvez le configurer SSM Agent lui-même ou configurer l' CloudWatch agent pour envoyer ces données de journal à Amazon CloudWatch Logs.

Important

L' CloudWatch agent unifié a été remplacé SSM Agent en tant qu'outil d'envoi des données de journal à Amazon CloudWatch Logs. Le plugin SSM Agent aws:cloudWatch n'est pas pris en charge. Nous vous recommandons de n'utiliser que l' CloudWatch agent unifié pour vos processus de collecte de journaux. Pour plus d'informations, consultez les rubriques suivantes :

- Envoi des journaux des nœuds vers CloudWatch des journaux unifiés (CloudWatch agent)
- Migrer la collecte des journaux des nœuds Windows Server vers l' CloudWatch agent
- Collecte de métriques, de journaux et de traces avec l' CloudWatch agent dans le guide de CloudWatch l'utilisateur Amazon.

Grâce CloudWatch aux journaux, vous pouvez surveiller les données des journaux en temps réel, rechercher et filtrer les données des journaux en créant un ou plusieurs filtres métriques, et archiver et récupérer les données historiques lorsque vous en avez besoin. Pour plus d'informations sur CloudWatch les journaux, consultez le guide de l'utilisateur Amazon CloudWatch Logs.

La configuration d'un agent pour envoyer des données de journal à Amazon CloudWatch Logs offre les avantages suivants :

- Stockage centralisé des fichiers journaux pour l'ensemble de vos fichiers journaux de l'SSM Agent.

- Accès plus rapide aux fichiers pour examiner les erreurs.
- Conservation des fichiers journaux illimitée (configurable).
- Les journaux peuvent être conservés et rester accessibles quel que soit l'état du nœud.
- Accès à d'autres CloudWatch fonctionnalités telles que les métriques et les alarmes.

Pour de plus amples informations sur la surveillance de l'activité Session Manager, consultez Auditer l'activité de session et Activation et désactivation de la journalisation des activités de session.

Migrer la collecte des journaux des nœuds Windows Server vers l' CloudWatch agent

Si vous utilisez l'un SSM Agent des Windows Server nœuds pris en charge pour envoyer des fichiers SSM Agent CloudWatch journaux à Amazon Logs, vous pouvez utiliser Systems Manager pour effectuer la migration depuis SSM Agent l' CloudWatch agent en tant qu'outil de collecte de journaux et migrer vos paramètres de configuration.

L' CloudWatch agent n'est pas pris en charge sur les versions 32 bits deWindows Server.

Pour les instances EC2 64 bits pourWindows Server, vous pouvez effectuer la migration vers l' CloudWatch agent 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 CloudWatch peuvent être interrompues ou dupliquées. Vos statistiques et vos données de journal seront à nouveau enregistrées avec précision une CloudWatch fois la migration terminée.

Nous vous recommandons de tester la migration sur un nombre limité de nœuds avant de migrer un parc complet vers l' CloudWatch agent. Après la migration, si vous préférez réaliser la collecte de journaux avec l'SSM Agent, vous pouvez reprendre son utilisation.

> **Important**
>
> Dans les cas suivants, vous ne pourrez pas migrer vers l' CloudWatch agent en suivant les étapes décrites dans cette rubrique :

- La configuration existante pour l'SSM Agent spécifie plusieurs régions.
- La configuration existante pour l'SSM Agent spécifie plusieurs jeux d'informations d'identification pour l'accès/la clé secrète.

Dans ces cas, il sera nécessaire de désactiver la collecte des journaux SSM Agent et d'installer l' CloudWatch agent sans processus de migration. Pour plus d'informations, consultez les rubriques suivantes du guide de CloudWatch l'utilisateur Amazon :

- Installation de l' CloudWatch agent
- Installation de l' CloudWatch agent sur des serveurs locaux

Avant de commencer

Avant de commencer une migration vers l' CloudWatch agent de collecte des journaux, assurez-vous que les nœuds sur lesquels vous allez effectuer la migration répondent aux exigences suivantes :

- Le système d'exploitation est une version 64 bits de Windows Server.
- SSM Agent 2.2.93.0 (ou version ultérieure) est installé sur le nœud.
- SSM Agent est configuré pour la surveillance sur le nœud.

Rubriques

Migration automatique vers l'agent CloudWatch

Pour les instances EC2 Windows Server uniquement, vous pouvez utiliser la AWS Systems Manager console ou le AWS Command Line Interface (AWS CLI) pour migrer automatiquement vers l' CloudWatch agent en tant qu'outil de collecte de journaux.

> **Note**
>
> AWS Systems Manager prend en charge la migration depuis SSM Agent l' CloudWatch agent unifié pour collecter des journaux et des métriques uniquement sur les versions 64

bits de Windows. Pour plus d'informations sur la configuration de l' CloudWatch agent unifié sur d'autres systèmes d'exploitation et pour des informations complètes sur l'utilisation de l' CloudWatch agent, consultez la section Collecte de métriques et de journaux à partir d'instances Amazon EC2 et de serveurs sur site avec l' CloudWatch agent dans le guide de l'utilisateur Amazon CloudWatch .
Vous pouvez utiliser l' CloudWatch agent 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 réussie, vérifiez vos résultats pour vous CloudWatch assurer que vous recevez les statistiques, les journaux ou les journaux d'événements Windows que vous attendez. Si vous êtes satisfait des résultats, vous pouvez éventuellement Stockez les paramètres de configuration de l' CloudWatch agent dans Parameter Store. Si la migration n'aboutit pas ou si les résultats ne sont pas ceux attendus, vous pouvez essayer Retourner à la collecte de journaux avec l'SSM Agent.

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. Supposons, par exemple, que l'`"LogStream": "{hostname}"`entrée suivante renvoie 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 est mappée vers 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 migrer automatiquement vers l' CloudWatch agent (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Run Command, puis Run command (Exécuter la commande).
3. Dans la liste Document de commande, sélectionnez `AmazonCloudWatch-MigrateCloudWatchAgent`.
4. Pour Statut, sélectionnez Enabled.
5. Dans la section Targets (Cibles), sélectionnez les nœuds gérés sur lesquels vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant des instances ou des appareils de périphérie manuellement ou en spécifiant un groupe de ressources.

   > **Tip**
   >
   > Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés pour obtenir des conseils de dépannage.

6. Pour Rate control (Contrôle de débit) :

   - Dans Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage de nœuds gérés sur lesquels exécuter simultanément la commande.

     > **Note**
     >
     > Si vous avez sélectionné des cibles en spécifiant les balises appliquées aux nœuds gérés ou en spécifiant des groupes de ressources AWS , et que vous n'êtes pas certain du nombre de nœuds gérés ciblés, limitez le nombre de cibles autorisées à exécuter simultanément le document en indiquant un pourcentage.

   - Dans Error threshold (Seuil d'erreur), indiquez quand arrêter l'exécution de la commande sur les autres nœuds gérés après l'échec de celle-ci sur un certain nombre ou un certain pourcentage de nœuds. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les nœuds gérés sur lesquels la commande est toujours en cours de traitement peuvent également envoyer des erreurs.

7. (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). Saisissez 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 (pour les instances EC2) ou de la fonction du service IAM (pour les machines activées par un système hybride) attribués à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, assurez-vous que le profil d'instance ou le rôle de service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

8. Dans la section SNS notifications (Notifications SNS), si vous souhaitez envoyer des notifications sur le statut d'exécution des commandes, cochez la case Enable SNS notifications (Activer les notifications SNS).

   Pour plus d'informations sur la configuration des notifications Amazon SNS pour Run Command, consultez Surveillance des changements d'état du Systems Manager à l'aide des notifications Amazon SNS.

9. Cliquez sur Exécuter.

Pour migrer automatiquement vers l' CloudWatch agent (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* représentent les ID des nœuds que vous souhaitez mettre à jour, par exemple i-02573cafcfEXAMPLE.

Migration manuelle vers l'agent CloudWatch

Pour les Windows Server nœuds sur site ou les instances EC2 pourWindows Server, suivez ces étapes pour migrer manuellement la collecte de journaux vers l'agent Amazon 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. Supposons, par exemple, que l'`"LogStream": "{hostname}"`entrée suivante renvoie 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 est mappée vers 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' CloudWatch agent (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Run Command, puis Run command (Exécuter la commande).
3. Dans la liste Document de commande, sélectionnez `AWS-ConfigureAWSPackage`.
4. Pour Action (Action) choisissez `Install`.
5. Pour Name (Nom), saisissez **`AmazonCloudWatchAgent`**.

6. Pour Version, saisissez **`latest`** si cette valeur n'est pas déjà fournie par défaut.
7. Dans la section Targets (Cibles), sélectionnez les nœuds gérés sur lesquels vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant des instances ou des appareils de périphérie manuellement ou en spécifiant un groupe de ressources.

   > **Tip**
   >
   > Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés pour obtenir des conseils de dépannage.

8. Pour Rate control (Contrôle de débit) :

   - Dans Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage de nœuds gérés sur lesquels exécuter simultanément la commande.

     > **Note**
     >
     > Si vous avez sélectionné des cibles en spécifiant les balises appliquées aux nœuds gérés ou en spécifiant des groupes de ressources AWS , et que vous n'êtes pas certain du nombre de nœuds gérés ciblés, limitez le nombre de cibles autorisées à exécuter simultanément le document en indiquant un pourcentage.

   - Dans Error threshold (Seuil d'erreur), indiquez quand arrêter l'exécution de la commande sur les autres nœuds gérés après l'échec de celle-ci sur un certain nombre ou un certain pourcentage de nœuds. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les nœuds gérés sur lesquels la commande est toujours 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). Saisissez 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 (pour les instances EC2) ou de la fonction du service IAM (pour les machines activées par un système hybride) attribués

à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, assurez-vous que le profil d'instance ou le rôle de service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

10. Dans la section SNS notifications (Notifications SNS), si vous souhaitez envoyer des notifications sur le statut d'exécution des commandes, cochez la case Enable SNS notifications (Activer les notifications SNS).

    Pour plus d'informations sur la configuration des notifications Amazon SNS pour Run Command, consultez Surveillance des changements d'état du Systems Manager à l'aide des notifications Amazon SNS.

11. Cliquez sur Exécuter.

Deux : Pour mettre à jour le format JSON des données de configuration

- Pour mettre à jour le format JSON des paramètres de configuration existants pour l' CloudWatch agent Run Command, utilisez une fonctionnalité du AWS Systems Manager nœud ou connectez-vous directement au nœud via une connexion RDP pour exécuter les PowerShell commandes Windows suivantes sur le nœud, une par une.

  ```
  cd ${Env:ProgramFiles}\\Amazon\\AmazonCloudWatchAgent
  ```

  ```
  .\\amazon-cloudwatch-agent-config-wizard.exe --isNonInteractiveWindowsMigration
  ```

  {Env :ProgramFiles} représente l'emplacement où se trouve généralement `C:\Program Files` le répertoire Amazon contenant l' CloudWatch agent.

Trois : pour configurer et démarrer l' CloudWatch agent (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Run Command, puis Run command (Exécuter la commande).

3. Dans la liste Document de commande, sélectionnez `AWS-RunPowerShellScript`.
4. Pour Commands (Commandes), saisissez 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 généralement `C:\Program Files` le répertoire Amazon contenant l' CloudWatch agent.

5. Dans la section Targets (Cibles), sélectionnez les nœuds gérés sur lesquels vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant des instances ou des appareils de périphérie manuellement ou en spécifiant un groupe de ressources.

> **Tip**
>
> Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés pour obtenir des conseils de dépannage.

6. Pour Rate control (Contrôle de débit) :

   - Dans Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage de nœuds gérés sur lesquels exécuter simultanément la commande.

   > **Note**
   >
   > Si vous avez sélectionné des cibles en spécifiant les balises appliquées aux nœuds gérés ou en spécifiant des groupes de ressources AWS , et que vous n'êtes pas certain du nombre de nœuds gérés ciblés, limitez le nombre de cibles autorisées à exécuter simultanément le document en indiquant un pourcentage.

   - Dans Error threshold (Seuil d'erreur), indiquez quand arrêter l'exécution de la commande sur les autres nœuds gérés après l'échec de celle-ci sur un certain nombre ou un certain pourcentage de nœuds. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les nœuds gérés sur lesquels la commande est toujours en cours de traitement peuvent également envoyer des erreurs.

7. (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). Saisissez 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 (pour les instances EC2) ou de la fonction du service IAM (pour les machines activées par un système hybride) attribués à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, assurez-vous que le profil d'instance ou le rôle de service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

8. Dans la section SNS notifications (Notifications SNS), si vous souhaitez envoyer des notifications sur le statut d'exécution des commandes, cochez la case Enable SNS notifications (Activer les notifications SNS).

   Pour plus d'informations sur la configuration des notifications Amazon SNS pour Run Command, consultez Surveillance des changements d'état du Systems Manager à l'aide des notifications Amazon SNS.

9. Cliquez sur Exécuter.

Quatre : Pour désactiver la collecte de journaux dans l'SSM Agent (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Run Command, puis Run command (Exécuter la commande).
3. Dans la liste Document de commande, sélectionnez `AWS-ConfigureCloudWatch`.
4. Pour Status (Statut), sélectionnez Disabled (Désactivé).

5. Dans la section Targets (Cibles), sélectionnez les nœuds gérés sur lesquels vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant des instances ou des appareils de périphérie manuellement ou en spécifiant un groupe de ressources.

   > **Tip**
   >
   > Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés pour obtenir des conseils de dépannage.

6. Pour Status (Statut), sélectionnez `Disabled`.
7. Pour Rate control (Contrôle de débit) :

   - Dans Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage de nœuds gérés sur lesquels exécuter simultanément la commande.

     > **Note**
     >
     > Si vous avez sélectionné des cibles en spécifiant les balises appliquées aux nœuds gérés ou en spécifiant des groupes de ressources AWS , et que vous n'êtes pas certain du nombre de nœuds gérés ciblés, limitez le nombre de cibles autorisées à exécuter simultanément le document en indiquant un pourcentage.

   - Dans Error threshold (Seuil d'erreur), indiquez quand arrêter l'exécution de la commande sur les autres nœuds gérés après l'échec de celle-ci sur un certain nombre ou un certain pourcentage de nœuds. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les nœuds gérés sur lesquels la commande est toujours en cours de traitement peuvent également envoyer des erreurs.
8. (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). Saisissez 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 (pour les instances EC2) ou de la fonction du service IAM (pour les machines activées par un système hybride) attribués

à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, assurez-vous que le profil d'instance ou le rôle de service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

9. Dans la section SNS notifications (Notifications SNS), si vous souhaitez envoyer des notifications sur le statut d'exécution des commandes, cochez la case Enable SNS notifications (Activer les notifications SNS).

   Pour plus d'informations sur la configuration des notifications Amazon SNS pour Run Command, consultez Surveillance des changements d'état du Systems Manager à l'aide des notifications Amazon SNS.

10. Cliquez sur Exécuter.

    Une fois ces étapes terminées, vérifiez vos connexions CloudWatch pour vérifier que vous recevez les métriques, les journaux ou les journaux d'événements Windows que vous attendez. Si vous êtes satisfait des résultats, vous pouvez éventuellement Stockez les paramètres de configuration de l' CloudWatch agent dans Parameter Store. 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'SSM Agent.

Stockez les paramètres de configuration de l' CloudWatch agent dans Parameter Store

Vous pouvez enregistrer le contenu d'un fichier de configuration d' CloudWatch agent dansParameter Store. Si vous conservez ces données de configuration dans un paramètre, plusieurs nœuds peuvent en tirer leurs paramètres de configuration, ce qui vous évite de créer ou de mettre à jour manuellement des fichiers de configuration sur vos nœuds. Par exemple, vous pouvez utiliser Run Command pour écrire le contenu du paramètre dans des fichiers de configuration sur plusieurs nœuds, ou utiliser State Manager une fonctionnalité permettant d' AWS Systems Manageréviter toute dérive de configuration dans les paramètres de configuration de l' CloudWatch agent sur un parc de nœuds.

Lorsque vous exécutez l'assistant de configuration de l' CloudWatch agent, vous pouvez choisir de le laisser enregistrer vos paramètres de configuration en tant que nouveau paramètre dansParameter

Store. Pour plus d'informations sur l'exécution de l'assistant de configuration de l' CloudWatch agent, consultez la section Création du fichier de configuration de l' CloudWatch agent avec l'assistant dans le guide de CloudWatch l'utilisateur Amazon.

Si vous avez exécuté l'assistant mais que vous n'avez pas choisi l'option permettant d'enregistrer les paramètres en tant que paramètre, ou si vous avez créé le fichier de configuration de l' CloudWatch agent manuellement, vous pouvez récupérer les données à enregistrer en tant que paramètre sur votre nœud dans le fichier suivant.

```
${Env:ProgramFiles}\Amazon\AmazonCloudWatchAgent\config.json
```

{Env :ProgramFiles} représente l'emplacement où se trouve généralement `C:\Program Files` le répertoire Amazon contenant l' CloudWatch agent.

Nous vous recommandons de conserver une sauvegarde du JSON de ce fichier à un emplacement autre que le nœud lui-même.

Pour plus d'informations sur la création d'un paramètre, consultez Création de paramètres Systems Manager.

Pour plus d'informations sur l' CloudWatch agent, consultez la section Collecte de métriques et de journaux à partir d'instances Amazon EC2 et de serveurs sur site avec l' CloudWatch agent dans le guide de l'utilisateur Amazon CloudWatch .

Retourner à la collecte de journaux avec l'SSM Agent

Si vous souhaitez utiliser à nouveau l'SSM Agent pour collecter les journaux, suivez les étapes décrites dans cette section.

Un : Pour récupérer les données de configuration à partir de l'SSM Agent

1. Sur le nœud où vous souhaitez recommencer à collecter les journaux à l'aide de SSM Agent, localisez le contenu du fichier de configuration de SSM Agent. Ce fichier JSON se trouve généralement à l'emplacement suivant :

   ```
   ${Env:ProgramFiles}\\Amazon\\SSM\\Plugins\\awsCloudWatch\
   \AWS.EC2.Windows.CloudWatch.json
   ```

   {Env :ProgramFiles} représente généralement `C:\Program Files` l'emplacement où se trouve le `Amazon` répertoire.

2. Copiez ces données dans un fichier texte pour les utiliser à une étape ultérieure.

   Nous vous recommandons de conserver une copie de sauvegarde du fichier JSON à un emplacement autre que le nœud lui-même.

Deux : pour désinstaller l' CloudWatch agent (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Run Command, puis Run command (Exécuter la commande).
3. Dans la liste Document de commande, sélectionnez `AWS-ConfigureAWSPackage`.
4. Dans Action (Action), choisissez Uninstall (Désinstaller).
5. Pour Name (Nom), saisissez **`AmazonCloudWatchAgent`**.
6. Dans la section Targets (Cibles), sélectionnez les nœuds gérés sur lesquels vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant des instances ou des appareils de périphérie manuellement ou en spécifiant un groupe de ressources.

   > **Tip**
   >
   > Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés pour obtenir des conseils de dépannage.

7. Pour Rate control (Contrôle de débit) :

   - Dans Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage de nœuds gérés sur lesquels exécuter simultanément la commande.

     > **Note**
     >
     > Si vous avez sélectionné des cibles en spécifiant les balises appliquées aux nœuds gérés ou en spécifiant des groupes de ressources AWS , et que vous n'êtes pas certain du nombre de nœuds gérés ciblés, limitez le nombre de cibles autorisées à exécuter simultanément le document en indiquant un pourcentage.

- Dans Error threshold (Seuil d'erreur), indiquez quand arrêter l'exécution de la commande sur les autres nœuds gérés après l'échec de celle-ci sur un certain nombre ou un certain pourcentage de nœuds. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les nœuds gérés sur lesquels la commande est toujours en cours de traitement peuvent également envoyer des erreurs.

8. (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). Saisissez 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 (pour les instances EC2) ou de la fonction du service IAM (pour les machines activées par un système hybride) attribués à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, assurez-vous que le profil d'instance ou le rôle de service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

9. Dans la section SNS notifications (Notifications SNS), si vous souhaitez envoyer des notifications sur le statut d'exécution des commandes, cochez la case Enable SNS notifications (Activer les notifications SNS).

   Pour plus d'informations sur la configuration des notifications Amazon SNS pour Run Command, consultez Surveillance des changements d'état du Systems Manager à l'aide des notifications Amazon SNS.

10. Cliquez sur Exécuter.

Trois : Pour réactiver la collecte de journaux dans SSM Agent (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.

2. Dans le panneau de navigation, sélectionnez Run Command, puis Run command (Exécuter la commande).
3. Dans la liste Document de commande, sélectionnez `AWS-ConfigureCloudWatch`.
4. Pour Status (Statut), sélectionnez `Enabled`.
5. Pour Properties (Propriétés), collez le contenu des anciennes données de configuration que vous avez enregistrées dans le fichier texte.
6. Dans la section Targets (Cibles), sélectionnez les nœuds gérés sur lesquels vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant des instances ou des appareils de périphérie manuellement ou en spécifiant un groupe de ressources.

   > **Tip**
   >
   > Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés pour obtenir des conseils de dépannage.

7. Pour Rate control (Contrôle de débit) :

   - Dans Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage de nœuds gérés sur lesquels exécuter simultanément la commande.

     > **Note**
     >
     > Si vous avez sélectionné des cibles en spécifiant les balises appliquées aux nœuds gérés ou en spécifiant des groupes de ressources AWS , et que vous n'êtes pas certain du nombre de nœuds gérés ciblés, limitez le nombre de cibles autorisées à exécuter simultanément le document en indiquant un pourcentage.

   - Dans Error threshold (Seuil d'erreur), indiquez quand arrêter l'exécution de la commande sur les autres nœuds gérés après l'échec de celle-ci sur un certain nombre ou un certain pourcentage de nœuds. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les nœuds gérés sur lesquels la commande est toujours en cours de traitement peuvent également envoyer des erreurs.

8. (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). Saisissez 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 (pour les instances EC2) ou de la fonction du service IAM (pour les machines activées par un système hybride) attribués à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, assurez-vous que le profil d'instance ou le rôle de service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

9. Dans la section SNS notifications (Notifications SNS), si vous souhaitez envoyer des notifications sur le statut d'exécution des commandes, cochez la case Enable SNS notifications (Activer les notifications SNS).

   Pour plus d'informations sur la configuration des notifications Amazon SNS pour Run Command, consultez Surveillance des changements d'état du Systems Manager à l'aide des notifications Amazon SNS.

10. Cliquez sur Exécuter.

Envoi de journaux SSM Agent à CloudWatch Logs

AWS Systems Manager Agent (SSM Agent) est un logiciel Amazon qui s'exécute sur les instances EC2, les appareils de périphérie, ainsi que les serveurs sur site et les machines virtuelles (VM) configurés pour Systems Manager. SSM Agent 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 SSM Agent, consultez Utilisation de l'option SSM Agent.

En outre, en suivant la procédure ci-après, vous pouvez configurer l'SSM Agent pour qu'il envoie les données des journaux à Amazon CloudWatch Logs.

Avant de commencer

Créer un groupe de journaux dans CloudWatch Logs. Pour plus d'informations, consultez Démarrer avec CloudWatch Logs dans le Guide de l'utilisateur Amazon CloudWatch Logs.

Pour configurer SSM Agent pour l'envoi de journaux à CloudWatch

1. Connectez-vous à un nœud et recherchez le fichier suivant :

   Linux

   Sur la plupart des types de nœuds 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`

   macOS

   `/opt/aws/ssm/seelog.xml.template`

   Windows

   `%ProgramFiles%\Amazon\SSM\seelog.xml.template`
2. Remplacez le nom du fichier `seelog.xml.template` par `seelog.xml`.

   > **Note**
   >
   > Sur Ubuntu Server 20.10 STR & 20.04, 18.04 et 16.04 LTS, le fichier `seelog.xml` doit être créé dans le répertoire `/etc/amazon/ssm/`. Vous pouvez créer ce répertoire et ce fichier en exécutant les commandes suivantes.
   >
   > ```
   > sudo mkdir -p /etc/amazon/ssm
   > ```
   >
   > ```
   > sudo cp -pr /snap/amazon-ssm-agent/current/* /etc/amazon/ssm
   > ```
   >
   > ```
   > sudo cp -p /etc/amazon/ssm/seelog.xml.template /etc/amazon/ssm/seelog.xml
   > ```
3. Ouvrez le fichier `seelog.xml` dans un éditeur de texte, puis localisez la section ci-après.

   Linux and macOS

   ```
   <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 and macOS

```
<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 SSM Agent ou le nœud.
6. Ouvrez la console CloudWatch à l'adresse https://console.aws.amazon.com/cloudwatch/.
7. Dans le volet de navigation, sélectionnez Log groups (Groupes de journaux), puis sélectionnez le nom de votre groupe de journaux.

> **Tip**
>
> Le flux de journaux des données des fichiers journaux SSM Agent est organisé par ID de nœud.

Surveillance de vos événements de demande de modification

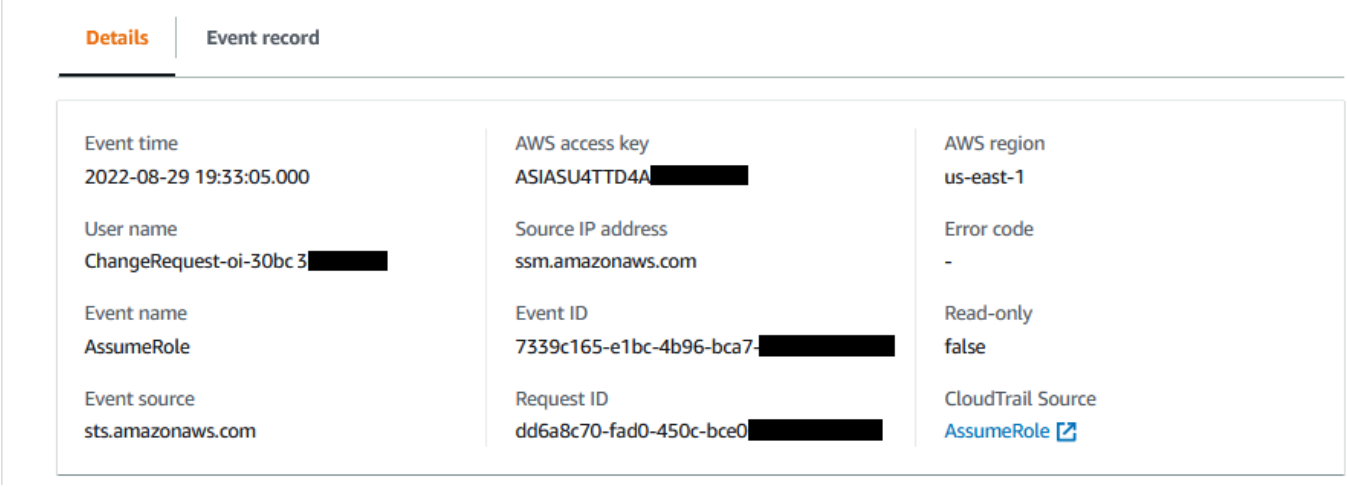
Après avoir activé l'intégration avec AWS CloudTrail Lake et créé un magasin de données d'événements, vous pouvez consulter des informations vérifiables sur les demandes de modification exécutées dans votre compte ou votre organisation. Cela inclut des détails tels que les suivants :

- L'identité de l'utilisateur à l'origine de la demande de modification
- L' Régions AWS endroit où les modifications ont été apportées
- L'adresse IP source de la demande
- La clé AWS d'accès utilisée pour la demande
- Les actions d'API exécutées pour la demande de modification
- Les paramètres de demande inclus pour ces actions
- Les ressources mises à jour au cours du processus

Vous trouverez ci-dessous des exemples de détails d'événements que vous pouvez consulter pour une demande de modification après avoir créé le magasin de données d'événements dans AWS CloudTrail Lake.

Details

L'image suivante montre les informations de haut niveau relatives à une demande de modification disponibles dans l'onglet Details (Détails). Ces informations incluent des données telles que l'heure à laquelle l'opération de demande de modification a commencé, l'ID de l'utilisateur qui a initié la demande de modification, l' Région AWS affectée, ainsi que les ID d'événement et de demande associés à la demande.



Event record

L'image suivante montre la structure du contenu JSON fourni par CloudTrail Lake pour un événement de demande de modification. Ces données sont fournies dans l'onglet Event record (Enregistrement d'événement) d'une demande de modification.

> **Important**
>
> Si vous utilisez Change Manager pour une organisation, vous pouvez réaliser la procédure suivante en étant connecté au compte de gestion ou au compte d'administrateur délégué de Change Manager.
>
> Toutefois, pour utiliser le compte d'administrateur délégué pour effectuer ces étapes, le même compte d'administrateur délégué doit être spécifié pour les deux CloudTrail etChange Manager.
>
> Lorsque vous vous connectez au compte de gestion pourChange Manager, vous pouvez ajouter ou modifier le compte d'administrateur délégué pour CloudTrail sur la page CloudTrail Paramètres. Cela doit être fait avant que le compte d'administrateur délégué ne puisse créer un magasin de données d'événements destiné à l'ensemble de l'organisation.

Pour activer le suivi des événements de CloudTrail Lake à partir de Change Manager

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Change Manager.

3. Sélectionnez l'onglet Requests (Demandes).
4. Choisissez n'importe quelle demande de modification existante, puis choisissez l'onglet Associated events (Événements associés).
5. Choisissez Enable CloudTrail Lake.
6. Suivez les étapes décrites dans la section Créer un magasin de données d' CloudTrail événements pour les événements du Guide de AWS CloudTrail l'utilisateur.

   Pour vous assurer que les données d'événement relatives à vos demandes de modification sont stockées, effectuez les sélections suivantes lors de la procédure :

   - Pour Type d'événement, laissez les AWS événements par défaut et les CloudTrailévénements sélectionnés.
   - Si vous utilisez Change Manager avec une organisation, sélectionnez Activer pour tous les comptes de mon organisation.
   - Pour Événements de gestion, ne décochez pas la case Écrire.

   Les autres options que vous choisissez lors de la création de votre magasin de données d'événements n'affectent pas le stockage des données d'événements pour vos demandes de modification.

Surveillance de vos automatisations

Les métriques sont les éléments de base d'Amazon 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.

Automation est une fonctionnalité de AWS Systems Manager. Systems Manager publie des métriques sur l'utilisation d'Automation sur CloudWatch. Cela vous permet de définir des alarmes en fonction de ces métriques.

Pour afficher les métriques Automation sur la console CloudWatch

1. Ouvrez la console CloudWatch à l'adresse https://console.aws.amazon.com/cloudwatch/.
2. Dans le panneau de navigation, sélectionnez Metrics (Métriques).
3. Sélectionnez SSM.

4. Sous l'onglet Metrics (Métriques), choisissez Usage (Utilisation), puis choisissez By AWS Resource.(Par ressource).
5. Dans la zone de recherche située près de la liste des métriques, saisissez SSM.

Pour afficher les métriques Automation à l'aide de la AWS CLI

Ouvrez une invite de commande et utilisez la commande suivante.

```
aws cloudwatch list-metrics \
    --namespace "AWS/Usage"
```

Métriques Automation

Systems Manager envoie les métriques Automation suivantes à CloudWatch.

Métrique	Description
`ConcurrentAutomationUsage`	Nombre d'automatisations exécutées en même temps dans le Compte AWS et la Région AWS actuels.
`QueuedAutomationUsage`	Nombre d'automatisations actuellement en file d'attente qui n'ont pas démarré et dont le statut est `Pending`.

Pour de plus amples informations sur l'utilisation de métriques CloudWatch, veuillez consulter les rubriques suivantes dans le Guide de l'utilisateur Amazon CloudWatch :

- Métriques
- Utilisation des métriques Amazon CloudWatch
- Utilisation d'alarmes Amazon CloudWatch

Surveillance des métriques Run Command avec Amazon CloudWatch

Les métriques sont les éléments de base d'Amazon 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 désormais des métriques sur le statut des commandes Run Command à CloudWatch, ce qui vous permet de définir des alarmes en fonction de ces métriques. Run Command une fonctionnalité de AWS Systems Manager. 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 du terminal pour les commandes pour lesquelles vous pouvez suivre les métriques incluent `Success`, `Failed` et `Delivery Timed Out`. Par exemple, pour un document SSM Command défini pour s'exécuter toutes les heures, vous pouvez configurer une alarme pour vous avertir lorsque le statut `Success` n'est pas signalé à une heure quelconque. Pour plus d'informations sur les valeurs de statut des commandes, consultez Comprendre les états des commandes.

Pour afficher les métriques sur la console CloudWatch

1. Ouvrez la console CloudWatch à l'adresse https://console.aws.amazon.com/cloudwatch/.
2. Dans le panneau de navigation, sélectionnez Metrics (Métriques).
3. Choisissez SSM-Run Command.dans la zone Alarms by AWS service (Alarmes de service), pour Services.

Pour afficher les métriques à l'aide de la AWS CLI

Ouvrez une invite de commande et utilisez la commande suivante.

```
aws cloudwatch list-metrics --namespace "AWS/SSM-RunCommand"
```

Pour répertorier toutes les rubriques, utilisez la commande suivante :

```
aws cloudwatch list-metrics
```

Métriques et dimensions de Systems Manager Run Command

Systems Manager envoie les métriques de commande Run Command à CloudWatch toutes les minutes.

Systems Manager envoie les métriques de commande suivantes à CloudWatch.

> **Note**
>
> 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 de plus amples informations sur l'utilisation de métriques CloudWatch, veuillez consulter les rubriques suivantes dans le Guide de l'utilisateur Amazon CloudWatch :

- Métriques
- Utilisation des métriques Amazon CloudWatch
- Utilisation d'alarmes Amazon CloudWatch

Journalisation des appels d' AWS Systems Manager API avec AWS CloudTrail

AWS Systems Manager est intégré à AWS CloudTrailun service qui fournit un enregistrement des actions entreprises par un utilisateur, un rôle ou un Service AWS. CloudTrail capture les appels

d'API pour Systems Manager sous forme d'événements. Les appels capturés incluent des appels provenant de la console Systems Manager et des appels de code vers les opérations de l'API Systems Manager. À l'aide des informations collectées par CloudTrail, vous pouvez déterminer la demande qui a été faite à Systems Manager, l'adresse IP à partir de laquelle la demande a été faite, la date à laquelle elle a été faite et des informations supplémentaires.

Chaque événement ou entrée de journal contient des informations sur la personne ayant initiée la demande.

- Utilisateur racine d'un compte AWS
- Informations d'identification de sécurité temporaires provenant d'un rôle AWS Identity and Access Management (IAM) ou d'un utilisateur fédéré.
- Informations d'identification de sécurité à long terme d'un utilisateur IAM.
- Demandes effectuées au nom d'un utilisateur de l'IAM Identity Center.
- Un autre Service AWS.

Pour plus d'informations, consultez l'élément CloudTrailUserIdentity.

CloudTrail est actif dans votre compte Compte AWS lorsque vous créez le compte et vous avez automatiquement accès à l'historique des CloudTrail événements. L'historique des CloudTrail événements fournit un enregistrement consultable, consultable, téléchargeable et immuable des 90 derniers jours des événements de gestion enregistrés dans un. Région AWS Pour plus d'informations, consultez la section Utilisation de l'historique des CloudTrail événements dans le guide de AWS CloudTrail l'utilisateur. La consultation de CloudTrail l'historique des événements est gratuite.

Pour un enregistrement continu des événements de vos 90 Compte AWS derniers jours, créez un magasin de données sur les événements de Trail ou CloudTrailLake.

CloudTrail sentiers

Un suivi permet CloudTrail de fournir des fichiers journaux à un compartiment Amazon S3. Tous les sentiers créés à l'aide du AWS Management Console sont multirégionaux. Vous pouvez créer un parcours à région unique ou multirégionale à l'aide du. AWS CLI Il est recommandé de créer un parcours multirégional, car vous capturez l'activité dans l'ensemble Régions AWS de votre compte. Si vous créez un parcours à région unique, vous ne pouvez voir que les événements enregistrés dans le parcours. Région AWS Pour plus d'informations sur les sentiers, consultez les

sections Création d'un sentier pour votre organisation Compte AWS et Création d'un sentier pour une organisation dans le guide de AWS CloudTrail l'utilisateur.

Vous pouvez envoyer une copie de vos événements de gestion en cours dans votre compartiment Amazon S3 gratuitement CloudTrail en créant un journal. Toutefois, des frais de stockage Amazon S3 sont facturés. Pour plus d'informations sur la CloudTrail tarification, consultez la section AWS CloudTrail Tarification. Pour obtenir des informations sur la tarification Amazon S3, consultez Tarification Amazon S3.

CloudTrail Stockages de données sur les événements du lac

CloudTrail Lake vous permet d'exécuter des requêtes SQL sur vos événements. CloudTrail Lake convertit les événements existants au format JSON basé sur les lignes au format Apache ORC. ORC est un format de stockage en colonnes qui est optimisé pour une récupération rapide des données. Les événements sont agrégés dans des magasins de données d'événement. Ceux-ci constituent des collections immuables d'événements basées sur des critères que vous sélectionnez en appliquant des sélecteurs d'événements avancés. Les sélecteurs que vous appliquez à un magasin de données d'événement contrôlent les événements qui persistent et que vous pouvez interroger. Pour plus d'informations sur CloudTrail Lake, consultez la section Travailler avec AWS CloudTrail Lake dans le guide de AWS CloudTrail l'utilisateur.

CloudTrail Les stockages et requêtes de données sur les événements de Lake entraînent des coûts. Lorsque vous créez un magasin de données d'événement, vous choisissez l'option de tarification que vous voulez utiliser pour le magasin de données d'événement. L'option de tarification détermine le coût d'ingestion et de stockage des événements, ainsi que les périodes de conservation par défaut et maximale pour le magasin de données d'événement. Pour plus d'informations sur la CloudTrail tarification, consultez la section AWS CloudTrail Tarification.

Événements liés aux données de Systems Manager dans CloudTrail

Les événements de données fournissent des informations sur les opérations de ressources effectuées sur ou dans une ressource (par exemple, la création ou l'ouverture d'un canal de contrôle). Ils sont également connus sous le nom opérations de plans de données. Les événements de données sont souvent des activités dont le volume est élevé. Par défaut, CloudTrail n'enregistre pas les événements liés aux données. L'historique des CloudTrail événements n'enregistre pas les événements liés aux données.

Des frais supplémentaires s'appliquent pour les événements de données. Pour plus d'informations sur la CloudTrail tarification, consultez la section AWS CloudTrail Tarification.

Vous pouvez enregistrer les événements de données pour les types de ressources Systems Manager à l'aide de la CloudTrail console ou AWS CLI des opérations de CloudTrail l'API. Pour plus d'informations sur la façon de consigner les événements liés aux données, consultez les sections Consignation des événements liés aux données avec le AWS Management Console et Enregistrement des événements liés aux données avec le AWS Command Line Interface dans le Guide de AWS CloudTrail l'utilisateur.

Le tableau suivant répertorie les types de ressources Systems Manager pour lesquels vous pouvez enregistrer des événements de données. La colonne Type d'événement de données (console) indique la valeur à choisir dans la liste des types d'événements de données de la CloudTrail console. La colonne de valeur resources.type indique la **`resources.type`** valeur que vous devez spécifier lors de la configuration de sélecteurs d'événements avancés à l'aide des API or. AWS CLI CloudTrail La CloudTrail colonne Data APIs logged to indique les appels d'API enregistrés CloudTrail pour le type de ressource.

Type d'événement de données (console)	valeur resources.type	API de données connectées à CloudTrail
Systems Manager	`AWS::SSMMessages::ControlChannel`	• `CreateControlChannel` • `OpenControlChannel` Pour plus d'informations sur ces opérations, consultez la section Actions définies par Amazon Message Gateway Service dans le Service Authorization Reference.
Nœud géré par Systems Manager	`AWS::SSM::ManagedNode`	• `RequestManagedInstanceRoleToken` — Cet événement est généré lorsque l'agent Systems Manager (agent SSM) exécuté sur un nœud géré par Systems Manager demande des informations

Type d'événement de données (console)	valeur resources.type	API de données connectées à CloudTrail
		d'identification au service d'identification Systems Manager.

Vous pouvez configurer des sélecteurs d'événements avancés pour filtrer les `eventName`, `readOnly`, et `resources.ARN` des champs pour enregistrer uniquement les événements importants pour vous. Pour plus d'informations sur ces champs, consultez AdvancedFieldSelectorla référence de l'AWS CloudTrail API.

Événements de gestion de Systems Manager dans CloudTrail

Les événements de gestion fournissent des informations sur les opérations de gestion effectuées sur les ressources de votre Compte AWS. Ils sont également connus sous le nom opérations de plan de contrôle. Par défaut, CloudTrail enregistre les événements de gestion.

Systems Manager enregistre toutes les opérations du plan de contrôle CloudTrail sous forme d'événements de gestion. Les opérations de l'API Systems Manager sont documentées dans la référence des AWS Systems Manager API. Par exemple, les appels aux `StartSession` actions `CreateMaintenanceWindows`,`PutInventory`,`SendCommand`, et génèrent des entrées dans les fichiers CloudTrail journaux. Pour un exemple de configuration CloudTrail pour surveiller un appel d'API Systems Manager, consultezSurveillance de l'activité des sessions à l'aide d'Amazon EventBridge (console).

Exemples d'événements Systems Manager

Un événement représente une demande unique provenant de n'importe quelle source et inclut des informations sur l'opération d'API demandée, la date et l'heure de l'opération, les paramètres de la demande, etc. CloudTrail les fichiers journaux ne constituent pas une trace ordonnée des appels d'API publics. Les événements n'apparaissent donc pas dans un ordre spécifique.

Exemples :

- Exemples d'événements de gestion
- Exemples d'événements liés aux données

Exemples d'événements de gestion

Exemple 1 : **`DeleteDocument`**

L'exemple suivant montre un CloudTrail événement qui illustre le `DeleteDocument` fonctionnement d'un document nommé `example-Document` dans la 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",
    "eventCategory": "Management"
}
```

Exemple 2 : **`StartConnection`**

L'exemple suivant montre un CloudTrail événement pour un utilisateur qui démarre une connexion RDP Fleet Manager en utilisant la région USA Est (Ohio) (us-east-2). L'action API sous-jacente est `StartConnection`.

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AKIAI44QH8DHBEXAMPLE",
        "arn": "arn:aws:sts::123456789012:assumed-role/exampleRole",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AKIAI44QH8DHBEXAMPLE",
                "arn": "arn:aws:sts::123456789012:assumed-role/exampleRole",
                "accountId": "123456789012",
                "userName": "exampleRole"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2021-12-13T14:57:05Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2021-12-13T16:50:41Z",
    "eventSource": "ssm-guiconnect.amazonaws.com",
    "eventName": "StartConnection",
    "awsRegion": "us-east-2",
    "sourceIPAddress": "34.230.45.60",
```

```
    "userAgent": "example-user-agent-string",
    "requestParameters": {
        "AuthType": "Credentials",
        "Protocol": "RDP",
        "ConnectionType": "SessionManager",
        "InstanceId": "i-02573cafcfEXAMPLE"
    },
    "responseElements": {
        "ConnectionArn": "arn:aws:ssm-guiconnect:us-east-2:123456789012:connection/
fcb810cd-241f-4aae-9ee4-02d59EXAMPLE",
        "ConnectionKey": "71f9629f-0f9a-4b35-92f2-2d253EXAMPLE",
        "ClientToken": "49af0f92-d637-4d47-9c54-ea51aEXAMPLE",
        "requestId": "d466710f-2adf-4e87-9464-055b2EXAMPLE"
    },
    "requestID": "d466710f-2adf-4e87-9464-055b2EXAMPLE",
    "eventID": "fc514f57-ba19-4e8b-9079-c2913EXAMPLE",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```

Exemples d'événements liés aux données

Exemple 1 : **`CreateControlChannel`**

L'exemple suivant montre un CloudTrail événement illustrant l'`CreateControlChannel`opération.

```
{
  "eventVersion":"1.08",
  "userIdentity":{
    "type":"AssumedRole",
    "principalId":"AKIAI44QH8DHBEXAMPLE",
    "arn":"arn:aws:sts::123456789012:assumed-role/exampleRole",
    "accountId":"123456789012",
    "accessKeyId":"AKIAI44QH8DHBEXAMPLE",
    "sessionContext":{
      "sessionIssuer":{
        "type":"Role",
        "principalId":"AKIAI44QH8DHBEXAMPLE",
        "arn":"arn:aws:iam::123456789012:role/exampleRole",
        "accountId":"123456789012",
```

```
        "userName":"exampleRole"
      },
      "attributes":{
        "creationDate":"2023-05-04T23:14:50Z",
        "mfaAuthenticated":"false"
      }
    }
  },
  "eventTime":"2023-05-04T23:53:55Z",
  "eventSource":"ssm.amazonaws.com",
  "eventName":"CreateControlChannel",
  "awsRegion":"us-east-1",
  "sourceIPAddress":"192.0.2.0",
  "userAgent":"example-agent",
  "requestParameters":{
    "channelId":"44295c1f-49d2-48b6-b218-96823EXAMPLE",
    "messageSchemaVersion":"1.0",
    "requestId":"54993150-0e8f-4142-aa54-3438EXAMPLE",
    "userAgent":"example-agent"
  },
  "responseElements":{
    "messageSchemaVersion":"1.0",
    "tokenValue":"Value hidden due to security reasons.",
    "url":"example-url"
  },
  "requestID":"54993150-0e8f-4142-aa54-3438EXAMPLE",
  "eventID":"a48a28de-7996-4ca1-a3a0-a51fEXAMPLE",
  "readOnly":false,
  "resources":[
    {
      "accountId":"123456789012",
      "type":"AWS::SSMMessages::ControlChannel",
      "ARN":"arn:aws:ssmmessages:us-east-1:123456789012:control-
channel/44295c1f-49d2-48b6-b218-96823EXAMPLE"
    }
  ],
  "eventType":"AwsApiCall",
  "managementEvent":false,
  "recipientAccountId":"123456789012",
  "eventCategory":"Data"
}
```


Exemple 2 : **RequestManagedInstanceRoleToken**

L'exemple suivant montre un CloudTrail événement illustrant l'`RequestManagedInstanceRoleToken`opération.


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "123456789012:aws:ec2-instance:i-02854e4bEXAMPLE",
        "arn": "arn:aws:sts::123456789012:assumed-role/aws:ec2-instance/
i-02854e4bEXAMPLE",
        "accountId": "123456789012",
        "accessKeyId": "AKIAI44QH8DHBEXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "123456789012:aws:ec2-instance",
                "arn": "arn:aws:iam::123456789012:role/aws:ec2-instance",
                "accountId": "123456789012",
                "userName": "aws:ec2-instance"
            },
            "attributes": {
                "creationDate": "2023-08-27T03:34:46Z",
                "mfaAuthenticated": "false"
            },
            "ec2RoleDelivery": "2.0"
        }
    },
    "eventTime": "2023-08-27T03:37:15Z",
    "eventSource": "ssm.amazonaws.com",
    "eventName": "RequestManagedInstanceRoleToken",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "Apache-HttpClient/UNAVAILABLE (Java/1.8.0_362)",
    "requestParameters": {
        "fingerprint": "i-02854e4bf85EXAMPLE"
    },
    "responseElements": null,
    "requestID": "2582cced-455b-4189-9b82-7b48EXAMPLE",
    "eventID": "7f200508-e547-4c27-982d-4da0EXAMLE",
    "readOnly": true,
    "resources": [
        {
            "accountId": "123456789012",
```

```
            "type": "AWS::SSM::ManagedNode",
            "ARN": "arn:aws:ec2:us-east-1:123456789012:instance/i-02854e4bEXAMPLE"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": false,
    "recipientAccountId": "123456789012",
    "eventCategory": "Data"
}
```

Pour plus d'informations sur le contenu des CloudTrail enregistrements, voir le contenu des CloudTrail enregistrements dans le Guide de AWS CloudTrail l'utilisateur.

Journalisation de la sortie d'actions Automation avec CloudWatch Logs

Automation, une fonctionnalité de AWS Systems Manager, s'intègre à Amazon CloudWatch Logs. Vous pouvez envoyer la sortie depuis des actions `aws:executeScript` dans vos runbooks vers le groupe de journaux que vous spécifiez. Systems Manager ne crée pas de groupe ou de flux de journaux pour les documents qui n'utilisent pas d'actions `aws:executeScript`. Si le document utilise `aws:executeScript`, la sortie envoyée à CloudWatch Logs concerne uniquement ces actions. Vous pouvez utiliser la sortie de l'action `aws:executeScript` stockée dans votre groupe de journaux CloudWatch Logs à des fins de débogage et de dépannage. Si vous sélectionnez un groupe de journaux chiffré, la sortie de l'action `aws:executeScript` est elle aussi chiffrée. La journalisation de la sortie d'actions `aws:executeScript` est un paramètre au niveau du compte.

Pour envoyer une sortie d'action à CloudWatch Logs pour les runbooks appartenant à Amazon, le rôle ou l'utilisateur qui exécute l'automatisation doit disposer d'une autorisation pour les opérations suivantes :

- `logs:CreateLogGroup`
- `logs:CreateLogStream`
- `logs:DescribeLogGroups`
- `logs:DescribeLogStreams`
- `logs:PutLogEvents`

Pour les runbooks que vous possédez, les mêmes autorisations doivent être ajoutées à la fonction du service IAM (ou AssumeRole) que vous utilisez pour exécuter le runbook.

Pour envoyer une sortie d'action à 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 de gauche, sélectionnez Automation (Automatisation).
3. Sélectionnez l'onglet Préférences, puis Modifier.
4. Cochez la case en regard de Envoyer la sortie à CloudWatch Logs.
5. (Recommandé) Cochez la case en regard de Chiffrer des données de journaux. Une fois cette option activée, les données de journaux sont chiffrées à l'aide du chiffrement côté serveur des clés spécifiées pour le groupe de journaux. Si vous ne voulez pas chiffrer les données de journaux qui sont envoyées à CloudWatch Logs, décochez la case. Décochez la case si le chiffrement n'est pas autorisé sur le groupe de journaux.
6. Groupe de journaux CloudWatch Logs : pour spécifier le groupe de journaux CloudWatch Logs existant dans votre Compte AWS, auquel vous voulez envoyer une sortie d'action, sélectionnez l'une des options suivantes :
   - Send output to the default log group (Envoyer la sortie au groupe de journaux par défaut) : si le groupe de journaux par défaut n'existe pas (`/aws/ssm/automation/executeScript`), Automation le crée pour vous.
   - Choose from a list of log groups (Choisir un groupe de journaux dans la liste) : utilisez un groupe de journaux qui a déjà été créé dans votre compte pour stocker les sorties d'action.
   - Enter a log group name (Saisir 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 sorties d'action.
7. Sélectionnez Enregistrer.

Pour envoyer une sortie d'action à CloudWatch Logs (ligne de commande)

1. Ouvrez votre outil de ligne de commande préféré et exécutez la commande suivante pour mettre à jour la destination de sortie de l'action.

Linux & macOS

```
aws ssm update-service-setting \
    --setting-id arn:aws:ssm:region:account-id:servicesetting/ssm/automation/
customer-script-log-destination \
    --setting-value CloudWatch
```

Windows

```
aws ssm update-service-setting ^
    --setting-id arn:aws:ssm:region:account-id:servicesetting/ssm/automation/
customer-script-log-destination ^
    --setting-value CloudWatch
```

PowerShell

```
Update-SSMServiceSetting `
    -SettingId "arn:aws:ssm:region:account-id:servicesetting/ssm/automation/
customer-script-log-destination" `
    -SettingValue "CloudWatch"
```

Il n'y a pas de sortie si la commande réussit.

2. Exécutez la commande suivante pour spécifier le groupe de journaux auquel vous voulez envoyer une sortie d'action.

Linux & macOS

```
aws ssm update-service-setting \
    --setting-id arn:aws:ssm:region:account-id:servicesetting/ssm/automation/
customer-script-log-group-name \
    --setting-value my-log-group
```

Windows

```
aws ssm update-service-setting ^
    --setting-id arn:aws:ssm:region:account-id:servicesetting/ssm/automation/
customer-script-log-group-name ^
    --setting-value my-log-group
```

PowerShell

```
Update-SSMServiceSetting `
    -SettingId "arn:aws:ssm:region:account-id:servicesetting/ssm/automation/
customer-script-log-group-name" `
    -SettingValue "my-log-group"
```

Il n'y a pas de sortie si la commande réussit.

3. Exécutez la commande suivante pour afficher les paramètres de service actuels pour les préférences de journalisation des actions Automation dans les Compte AWS et Région AWS actuels.

Linux & macOS

```
aws ssm get-service-setting \
    --setting-id arn:aws:ssm:region:account-id:servicesetting/ssm/automation/
customer-script-log-destination
```

Windows

```
aws ssm get-service-setting ^
    --setting-id arn:aws:ssm:region:account-id:servicesetting/ssm/automation/
customer-script-log-destination
```

PowerShell

```
Get-SSMServiceSetting `
    -SettingId "arn:aws:ssm:region:account-id:servicesetting/ssm/automation/
customer-script-log-destination"
```

La commande renvoie des informations telles que les suivantes.

```
{
    "ServiceSetting": {
        "Status": "Customized",
        "LastModifiedDate": 1613758617.036,
        "SettingId": "/ssm/automation/customer-script-log-destination",
```

```
        "LastModifiedUser": "arn:aws:sts::123456789012:assumed-role/Administrator/
User_1",
        "SettingValue": "CloudWatch",
        "ARN": "arn:aws:ssm:us-east-2:123456789012:servicesetting/ssm/automation/
customer-script-log-destination"
    }
}
```

Configuration d'Amazon CloudWatch Logs pour Run Command

Lorsque vous envoyez une commande à l'aide Run Command d'une fonctionnalité de AWS Systems Manager, vous pouvez spécifier où vous souhaitez envoyer le résultat de la commande. Par défaut, Systems Manager renvoie uniquement les 24 000 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). Vous pouvez également spécifier Amazon CloudWatch Logs. Si vous spécifiez CloudWatch Logs, envoie Run Command régulièrement toutes les sorties de commande et tous 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 nœud géré pour utiliser les politiques `AmazonSSMManagedInstanceCore` gérées AWS Identity and Access Management (IAM)`CloudWatchAgentServerPolicy`, votre nœud ne nécessite aucune configuration supplémentaire pour envoyer des résultats à CloudWatch Logs. Choisissez cette option si vous envoyez des commandes depuis la console, ou ajoutez la `cloud-watch-output-config` section et le `CloudWatchOutputEnabled` paramètre si vous utilisez le AWS Command Line Interface (AWS CLI) ou une opération d'API. AWS Tools for Windows PowerShell La section `cloud-watch-output-config` et le paramètre `CloudWatchOutputEnabled` sont décrits plus en détail ultérieurement dans cette rubrique.

Pour plus d'informations sur l'ajout de politiques à un profil d'instance pour les instances EC2, consultez Configurer les autorisations d'instance requises pour Systems Manager. Pour plus d'informations sur l'ajout de politiques à un rôle de service pour les serveurs locaux et les machines virtuelles que vous prévoyez d'utiliser en tant que nœuds gérés, consultez la section Créer le rôle de service IAM requis pour Systems Manager dans les environnements hybrides et multicloud.

Si vous utilisez une politique personnalisée sur vos nœuds, mettez-la à jour pour chaque nœud afin de permettre à Systems Manager d'envoyer des résultats et des CloudWatch journaux à

Logs. Ajoutez les objets de politique suivants à votre politique personnalisée. Pour de plus amples informations sur la mise à jour d'une politique IAM, consultez Modification de politiques IAM dans le Guide de l'utilisateur IAM.

```
{
    "Effect": "Allow",
    "Action": "logs:DescribeLogGroups",
    "Resource": "*"
},
{
   "Effect":"Allow",
   "Action":[
      "logs:CreateLogGroup",
      "logs:CreateLogStream",
      "logs:DescribeLogStreams",
      "logs:PutLogEvents"
   ],
   "Resource":"arn:aws:logs:*:*:log-group:/aws/ssm/*"
},
```

Spécification CloudWatch des journaux lorsque vous envoyez des commandes

Pour spécifier CloudWatch les journaux comme sortie lorsque vous envoyez une commande depuis le AWS Management Console, choisissez CloudWatch Sortie dans la section Options de sortie. Vous pouvez éventuellement 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 est nommé selon le format : `/aws/ssm/`*`SystemsManagerDocumentName`*.

Si vous exécutez des commandes à l'aide du AWS CLI, spécifiez la `cloud-watch-output-config` section dans votre commande. Cette section vous permet de spécifier le paramètre `CloudWatchOutputEnabled`, et éventuellement le paramètre `CloudWatchLogGroupName`. Voici un exemple.

Linux & macOS

```
aws ssm send-command \
    --instance-ids "instance ID" \
    --document-name "AWS-RunShellScript" \
```

```
    --parameters "commands=echo helloWorld" \
    --cloud-watch-output-config
 "CloudWatchOutputEnabled=true,CloudWatchLogGroupName=log group name"
```

Windows

```
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 des commandes dans CloudWatch les journaux

Dès que la commande commence à s'exécuter, Systems Manager envoie des résultats à CloudWatch Logs en temps quasi réel. La sortie dans CloudWatch Logs utilise le 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 commande telle qu'elle est affichée 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 d'événements Systems Manager avec Amazon EventBridge

Amazon EventBridge est un service de bus d'événements sans serveur que vous pouvez utiliser pour connecter vos applications à des données provenant de diverses sources. EventBridge fournit un flux de données en temps réel à partir de vos propres applications, d'applications logiciel en tant que service (SaaS) et de Services AWS, puis achemine ces données vers des cibles telles que AWS Lambda. Vous pouvez configurer des règles de routage afin de déterminer la cible des données permettant de créer des architectures applicatives capables de réagir en temps réel à l'ensemble de vos sources de données. EventBridge vous permet de créer des architectures orientées événement, vaguement couplées et distribuées.

Auparavant, EventBridge s'appelait Amazon CloudWatch Events. EventBridge 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. Comme EventBridge utilise la même API CloudWatch Events, l'utilisation de votre API CloudWatch Events existante ne change pas.

EventBridge peut ajouter à vos règles des événements provenant de douzaines de Services AWS et des cibles provenant de plus de 20 Services AWS.

EventBridge prend à la fois en charge des événements AWS Systems Manager et des cibles Systems Manager.

Types d'événement Systems Manager pris en charge

EventBridge peut détecter de nombreux types d'événements Systems Manager, parmi lesquels :

- Une fenêtre de maintenance désactivée.
- Un flux de travail Automation terminé avec succès. Automation est une fonctionnalité de AWS Systems Manager.
- Un nœud géré non conforme aux correctifs.
- Une valeur de paramètre mise à jour.

EventBridge prend en charge des événements provenant des fonctionnalités AWS Systems Manager suivantes :

- Automatisation (Les événements sont générés sur la base du meilleur effort.)
- Change Calendar (Les événements sont générés sur la base du meilleur effort.)
- Conformité d'
- Inventory (Les événements sont générés sur la base du meilleur effort.)
- Maintenance Windows (Les événements sont générés sur la base du meilleur effort.)
- Parameter Store (Les événements sont générés sur la base du meilleur effort.)
- Run Command (Les événements sont générés sur la base du meilleur effort.)
- State Manager (Les événements sont générés sur la base du meilleur effort.)

Pour des détails complets sur les types d'événements Systems Manager pris en charge, veuillez consulter Référence : modèles et types d'événements Amazon EventBridge pour Systems Manager et Exemples EventBridge d'événements Amazon pour Systems Manager.

Types de cibles Systems Manager prises en charge

EventBridge prend en charge les trois fonctionnalités Systems Manager suivantes en tant que cibles d'une règle d'événement :

- Exécution d'un flux de travail Automation
- Exécution d'un document de commande Run Command (Les événements sont générés dans la mesure du possible.)
- Création d'un OpsItem OpsCenter

Pour obtenir des suggestions sur l'utilisation de ces cibles, veuillez consulter Exemples de scénarios : cibles Systems Manager dans les règles Amazon EventBridge.

Pour plus d'informations sur la mise en route avec EventBridge et la configuration de règles, consultez Démarrage avec Amazon EventBridge dans le Guide de l'utilisateur Amazon EventBridge. Pour obtenir des informations détaillées sur l'utilisation d'EventBridge, veuillez consulter le Guide de l'utilisateur Amazon EventBridge.

Rubriques

- Configurer EventBridge pour des événements Systems Manager
- Exemples EventBridge d'événements Amazon pour Systems Manager
- Exemples de scénarios : cibles Systems Manager dans les règles Amazon EventBridge

Configurer EventBridge pour des événements Systems Manager

Utilisez Amazon EventBridge pour générer un événement cible lorsque des changements de statut AWS Systems Manager, des changements d'état ou d'autres conditions se produisent. Vous pouvez créer une règle qui s'exécute à chaque changement d'état ou de statut, ou lorsqu'un ou plusieurs statuts spécifiques sont activés.

La procédure suivante décrit les étapes générales de création d'une règle EventBridge qui s'applique lorsqu'un événement spécifié est généré par Systems Manager. Pour obtenir la liste des procédures qui traitent des scénarios spécifiques dans ce guide de l'utilisateur, veuillez consulter Plus d'informations à la fin de cette rubrique.

> **Note**
>
> Lorsqu'un service de votre Compte AWS émet un événement, il accède toujours au bus d'événement par défaut de votre compte. Pour écrire une règle qui se déclenche sur des événements provenant de Services AWS de 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 les événements des Services AWS, mais cette règle se déclenche uniquement lorsque vous recevez un tel événement d'un autre compte à travers la livraison d'événements entre comptes. Pour plus d'informations, consultez Envoi et réception d'événements EventBridge entre Comptes AWS dans le Guide de l'utilisateur Amazon EventBridge.

Pour configurer EventBridge pour des événements 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.
3. Choisissez Create rule.
4. 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 AWS et sur le même bus d'événement.
5. Pour Event bus (Bus d'événement), sélectionnez le bus d'événement que vous souhaitez associer à cette règle. Si vous souhaitez que cette règle s'applique aux événements correspondants provenant de votre propre Compte AWS, sélectionnez défaut. Lorsqu'un Service AWS de votre compte émet un événement, il accède toujours au bus d'événement par défaut de votre compte.

6. Pour Rule type (Type de règle), choisissez Rule with an event pattern (Règle avec un modèle d'événement).
7. Choisissez Next (Suivant).
8. Pour Event source (Origine de l'événement), choisissez events or EventBridge partner events (Événements AWS ou événements partenaires EventBridge).
9. Dans la section Event pattern (Modèle d'événement), choisissez Event pattern form (Modèle d'événement).
10. Pour Event source (Origine de l'événement), choisissez AWSservices (Services).
11. Pour le AWS service choisissez Systems Manager.
12. Pour Event type (Type d'événement), effectuez l'une des actions suivantes :

    - Sélectionnez All Events (Tous les événements).

      Si vous sélectionnez All Events (Tous les événements), tous les événements émis par Systems Manager correspondent à la règle. Sachez que cette option peut entraîner de nombreuses actions de cible d'événement.
    - Sélectionnez le type d'événement Systems Manager à utiliser pour cette règle. EventBridge prend en charge des événements provenant des fonctionnalités AWS Systems Manager suivantes :
      - Automatisation
      - Change Calendar
      - Conformité d'
      - Inventory
      - Maintenance Windows
      - Parameter Store
      - Run Command
      - State Manager

      > **Note**
      >
      > Pour les actions Systems Manager non prises en charge par EventBridge, vous pouvez choisir un appel d'API AWS via CloudTrail pour créer une règle d'événement basée sur un appel d'API, qui sont enregistrées par CloudTrail. Pour voir un exemple,

consultez Surveillance de l'activité des sessions à l'aide d'Amazon EventBridge (console).

13. (Facultatif) Pour rendre la règle plus précise, ajoutez des valeurs de filtre. Par exemple, si vous avez choisi la région State Manager et souhaitez limiter la règle à l'état d'une instance gérée unique ciblée par une association, pour Specific type(s) (Type(s) spécifique(s)), choisissez EC2 State Manager Instance Association State Change (Changements de l'état d'association d'instance dans EC2 State Manager).

    Pour plus d'informations sur les types de détails pris en charge, consultez Référence : modèles et types d'événements Amazon EventBridge pour Systems Manager.

    Certains types de détails prennent en charge d'autres options comme le statut. Les options disponibles dépendent de la fonctionnalité que vous avez sélectionnée.

14. Choisissez Next (Suivant).
15. Pour Types de cibles, choisissez service AWS.
16. Pour Select a target (Sélectionnez une cible), choisissez une cible comme rubrique Amazon SNS ou une fonction AWS Lambda. La cible est déclenchée lorsqu'un événement correspond au modèle d'événement défini dans la règle est reçu.
17. Pour de nombreux types de cibles, EventBridge a besoin d'autorisations pour envoyer des événements à la cible. Dans ce cas, EventBridge peut créer le rôle AWS Identity and Access Management (IAM) nécessaire à l'exécution de votre règle :

    - Pour créer un rôle IAM automatiquement, sélectionnez Create a new role for this specific resource.
    - Pour utiliser un rôle IAM que vous avez créé auparavant, sélectionnez Use existing role (Utiliser un rôle existant).

18. (Facultatif) Sélectionnez Add another target (Ajouter une autre cible) pour ajouter une nouvelle cible pour cette règle.
19. Choisissez Next (Suivant).
20. (Facultatif) Saisissez une ou plusieurs balises pour la règle. Pour plus d'informations, consultez Balises Amazon EventBridge dans le Guide de l'utilisateur Amazon EventBridge.
21. Choisissez Next (Suivant).
22. Consultez les détails de la règle et choisissez Create rule (Créer une règle).

Plus d'informations

- Création d'un EventBridge événement utilisant un runbook (console)
- Transmission de données à Automation à l'aide de transformateurs en entrée
- Résolution des problèmes de conformité avec EventBridge
- Affichage des actions de suppression d'inventaire dans EventBridge
- Configurer des règles EventBridge pour créer des OpsItems
- Configuration des EventBridge règles pour les paramètres et des politiques de paramètres

Exemples EventBridge d'événements Amazon pour Systems Manager

Voici des exemples, au format JSON, d' EventBridge événements pris en charge pour AWS Systems Manager.

Types d'événement Systems Manager

- AWS Systems Manager Événements d'automatisation
- AWS Systems Manager EvénementsChange Calendar
- AWS Systems Manager EvénementsChange Manager
- AWS Systems Manager Événements relatifs à la conformité
- AWS Systems Manager EvénementsMaintenance Windows
- AWS Systems Manager EvénementsParameter Store
- AWS Systems Manager EvénementsOpsCenter
- AWS Systems Manager EvénementsRun Command
- AWS Systems Manager EvénementsState Manager

AWS Systems Manager Événements d'automatisation

Notification de changement de statut d'une étape d'automatisation

```
{
  "version": "0",
  "id": "eeca120b-a321-433e-9635-dab369006a6b",
  "detail-type": "EC2 Automation Step Status-change Notification",
  "source": "aws.ssm",
```

```
  "account": "123456789012",
  "time": "2016-11-29T19:43:35Z",
  "region": "us-east-1",
  "resources": ["arn:aws:ssm:us-east-2:123456789012:automation-
execution/333ba70b-2333-48db-b17e-a5e69c6f4d1c",
    "arn:aws:ssm:us-east-2:123456789012:automation-definition/runcommand1:1"],
  "detail": {
    "ExecutionId": "333ba70b-2333-48db-b17e-a5e69c6f4d1c",
    "Definition": "runcommand1",
    "DefinitionVersion": 1.0,
    "Status": "Success",
    "EndTime": "Nov 29, 2016 7:43:25 PM",
    "StartTime": "Nov 29, 2016 7:43:23 PM",
    "Time": 2630.0,
    "StepName": "runFixedCmds",
    "Action": "aws:runCommand"
  }
}
```

Notification de changement de statut d'exécution de l'automatisation

```
{
  "version": "0",
  "id": "d290ece9-1088-4383-9df6-cd5b4ac42b99",
  "detail-type": "EC2 Automation Execution Status-change Notification",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2016-11-29T19:43:35Z",
  "region": "us-east-2",
  "resources": ["arn:aws:ssm:us-east-2:123456789012:automation-
execution/333ba70b-2333-48db-b17e-a5e69c6f4d1c",
    "arn:aws:ssm:us-east-2:123456789012:automation-definition/runcommand1:1"],
  "detail": {
    "ExecutionId": "333ba70b-2333-48db-b17e-a5e69c6f4d1c",
    "Definition": "runcommand1",
    "DefinitionVersion": 1.0,
    "Status": "Success",
    "StartTime": "Nov 29, 2016 7:43:20 PM",
    "EndTime": "Nov 29, 2016 7:43:26 PM",
    "Time": 5753.0,
    "ExecutedBy": "arn:aws:iam::123456789012:user/userName"
  }
}
```

AWS Systems Manager EvénementsChange Calendar

Vous trouverez ci-dessous des exemples d'événements pour AWS Systems ManagerChange Calendar.

> **Note**
>
> Les modifications d'état des calendriers partagés depuis d'autres utilisateurs ne Comptes AWS sont actuellement pas prises en charge.

Calendrier OPEN (OUVERT)

```
{
    "version": "0",
    "id": "47a3f03a-f30d-1011-ac9a-du3bdEXAMPLE",
    "detail-type": "Calendar State Change",
    "source": "aws.ssm",
    "account": "123456789012",
    "time": "2020-09-19T18:00:07Z",
    "region": "us-east-2",
    "resources": [
        "arn:aws:ssm:us-east-2:123456789012:document/MyCalendar"
    ],
    "detail": {
        "state": "OPEN",
        "atTime": "2020-09-19T18:00:07Z",
        "nextTransitionTime": "2020-10-11T18:00:07Z"
    }
}
```

Calendrier CLOSED (FERMÉ)

```
{
    "version": "0",
    "id": "f30df03a-1011-ac9a-47a3-f761eEXAMPLE",
    "detail-type": "Calendar State Change",
    "source": "aws.ssm",
    "account": "123456789012",
    "time": "2020-09-17T21:40:02Z",
    "region": "us-east-2",
    "resources": [
```

```
        "arn:aws:ssm:us-east-2:123456789012:document/MyCalendar"
    ],
    "detail": {
        "state": "CLOSED",
        "atTime": "2020-08-17T21:40:00Z",
        "nextTransitionTime": "2020-09-19T18:00:07Z"
    }
}
```

AWS Systems Manager EvénementsChange Manager

Notification de mise à jour du statut d'une demande de modification : exemple 1

```
{
  "version": "0",
  "id": "feab80c1-a8ff-c721-b8b1-96ce70939696",
  "detail-type": "Change Request Status Update",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2023-10-24T10:51:52Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ssm:us-west-2:123456789012:opsitem/oi-12345abcdef",
    "arn:aws:ssm:us-west-2:123456789012:document/MyRunbook1"
  ],
  "detail": {
    "change-request-id": "d0585556-80f6-4522-8dad-dada6d45b67d",
    "change-request-title": "A change request title",
    "ops-item-id": "oi-12345abcdef",
    "ops-item-created-by": "arn:aws:iam::123456789012:user/JohnDoe",
    "ops-item-created-time": "2023-10-24T10:50:33.180334Z",
    "ops-item-modified-by": "arn:aws:iam::123456789012:user/JohnDoe",
    "ops-item-modified-time": "2023-10-24T10:50:33.180340Z",
    "ops-item-status": "InProgress",
    "change-template-document-name": "MyChangeTemplate",
    "runbook-document-arn": "arn:aws:ssm:us-west-2:123456789012:document/MyRunbook1",
    "runbook-document-version": "1",
    "auto-approve": true,
    "approvers": [
      "arn:aws:iam::123456789012:user/JaneDoe"
    ]
  }
}
```

Notification de mise à jour du statut d'une demande de modification : exemple 2


```
{
  "version": "0",
  "id": "25ce6b03-2e4e-1a2b-2a8f-6c9de8d278d2",
  "detail-type": "Change Request Status Update",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2023-10-24T10:51:52Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ssm:us-west-2:123456789012:opsitem/oi-abcdef12345",
    "arn:aws:ssm:us-west-2:123456789012:document/MyRunbook1"
  ],
  "detail": {
    "change-request-id": "d0585556-80f6-4522-8dad-dada6d45b67d",
    "change-request-title": "A change request title",
    "ops-item-id": "oi-abcdef12345",
    "ops-item-created-by": "arn:aws:iam::123456789012:user/JohnDoe",
    "ops-item-created-time": "2023-10-24T10:50:33.180334Z",
    "ops-item-modified-by": "arn:aws:iam::123456789012:user/JohnDoe",
    "ops-item-modified-time": "2023-10-24T10:50:33.997163Z",
    "ops-item-status": "Rejected",
    "change-template-document-name": "MyChangeTemplate",
    "runbook-document-arn": "arn:aws:ssm:us-west-2:123456789012:document/MyRunbook1",
    "runbook-document-version": "1",
    "auto-approve": true,
    "approvers": [
      "arn:aws:iam::123456789012:user/JaneDoe"
    ]
  }
}
```


AWS Systems Manager Événements relatifs à la conformité

Vous trouverez ci-dessous des exemples d'événements relatifs à la AWS Systems Manager conformité.

Conforme à l'association


```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
```

```
  "detail-type": "Configuration Compliance State Change",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2017-07-17T19:03:26Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:ssm:us-east-2:123456789012:managed-instance/i-01234567890abcdef"
  ],
  "detail": {
    "last-runtime": "2017-01-01T10:10:10Z",
    "compliance-status": "compliant",
    "resource-type": "managed-instance",
    "resource-id": "i-01234567890abcdef",
    "compliance-type": "Association"
  }
}
```

Non conforme à l'association

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Configuration Compliance State Change",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2017-07-17T19:02:31Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:ssm:us-east-2:123456789012:managed-instance/i-01234567890abcdef"
  ],
  "detail": {
    "last-runtime": "2017-01-01T10:10:10Z",
    "compliance-status": "non_compliant",
    "resource-type": "managed-instance",
    "resource-id": "i-01234567890abcdef",
    "compliance-type": "Association"
  }
}
```

Conforme aux correctifs

```
{
  "version": "0",
```

```
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Configuration Compliance State Change",
  "source": "aws.123456789012",
  "account": "123456789012",
  "time": "2017-07-17T19:03:26Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:ssm:us-east-2:123456789012:managed-instance/i-01234567890abcdef"
  ],
  "detail": {
    "resource-type": "managed-instance",
    "resource-id": "i-01234567890abcdef",
    "compliance-status": "compliant",
    "compliance-type": "Patch",
    "patch-baseline-id": "PB789",
    "severity": "critical"
  }
}
```

Non conforme aux correctifs

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "Configuration Compliance State Change",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2017-07-17T19:02:31Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:ssm:us-east-2:123456789012:managed-instance/i-01234567890abcdef"
  ],
  "detail": {
    "resource-type": "managed-instance",
    "resource-id": "i-01234567890abcdef",
    "compliance-status": "non_compliant",
    "compliance-type": "Patch",
    "patch-baseline-id": "PB789",
    "severity": "critical"
  }
}
```

AWS Systems Manager EvénementsMaintenance Windows

Vous trouverez ci-dessous des exemples d'événements pour Systems Manager Maintenance Windows.

Enregistrer une cible

L'autre valeur d'état valide est `DEREGISTERED`.

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-0123456789ab",
   "detail-type":"Maintenance Window Target Registration Notification",
   "source":"aws.ssm",
   "account":"123456789012",
   "time":"2016-11-16T00:58:37Z",
   "region":"us-east-2",
   "resources":[
      "arn:aws:ssm:us-east-2:123456789012:maintenancewindow/mw-0ed7251d3fcf6e0c2",
      "arn:aws:ssm:us-east-2:123456789012:windowtarget/
e7265f13-3cc5-4f2f-97a9-7d3ca86c32a6"
   ],
   "detail":{
      "window-target-id":"e7265f13-3cc5-4f2f-97a9-7d3ca86c32a6",
      "window-id":"mw-0ed7251d3fcf6e0c2",
      "status":"REGISTERED"
   }
}
```

Type d'exécution de fenêtre

Les autres valeurs d'état valides sont `PENDING`, `IN_PROGRESS`, `SUCCESS`, `FAILED`, `TIMED_OUT` et `SKIPPED_OVERLAPPING`.

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-0123456789ab",
   "detail-type":"Maintenance Window Execution State-change Notification",
   "source":"aws.ssm",
   "account":"123456789012",
   "time":"2016-11-16T01:00:57Z",
   "region":"us-east-2",
```

```
   "resources":[
      "arn:aws:ssm:us-east-2:123456789012:maintenancewindow/mw-123456789012345678"
   ],
   "detail":{
      "start-time":"2016-11-16T01:00:56.427Z",
      "end-time":"2016-11-16T01:00:57.070Z",
      "window-id":"mw-0ed7251d3fcf6e0c2",
      "window-execution-id":"b60fb56e-776c-4e5c-84ee-123456789012",
      "status":"TIMED_OUT"
   }
}
```

Type d'exécution de tâche

Les autres valeurs d'état valides sont `IN_PROGRESS`, `SUCCESS`, `FAILED` et `TIMED_OUT`.

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-0123456789ab",
   "detail-type":"Maintenance Window Task Execution State-change Notification",
   "source":"aws.ssm",
   "account":"123456789012",
   "time":"2016-11-16T01:00:56Z",
   "region":"us-east-2",
   "resources":[
      "arn:aws:ssm:us-east-2:123456789012:maintenancewindow/mw-123456789012345678"
   ],
   "detail":{
      "start-time":"2016-11-16T01:00:56.759Z",
      "task-execution-id":"6417e808-7f35-4d1a-843f-123456789012",
      "end-time":"2016-11-16T01:00:56.847Z",
      "window-id":"mw-0ed7251d3fcf6e0c2",
      "window-execution-id":"b60fb56e-776c-4e5c-84ee-123456789012",
      "status":"TIMED_OUT"
   }
}
```

Cible de tâche traitée

Les autres valeurs d'état valides sont `IN_PROGRESS`, `SUCCESS`, `FAILED` et `TIMED_OUT`.

```
{
```

```
   "version":"0",
   "id":"01234567-0123-0123-0123-0123456789ab",
   "detail-type":"Maintenance Window Task Target Invocation State-change Notification",
   "source":"aws.ssm",
   "account":"123456789012",
   "time":"2016-11-16T01:00:57Z",
   "region":"us-east-2",
   "resources":[
      "arn:aws:ssm:us-east-2:123456789012:maintenancewindow/mw-123456789012345678"
   ],
   "detail":{
      "start-time":"2016-11-16T01:00:56.427Z",
      "end-time":"2016-11-16T01:00:57.070Z",
      "window-id":"mw-0ed7251d3fcf6e0c2",
      "window-execution-id":"b60fb56e-776c-4e5c-84ee-123456789012",
      "task-execution-id":"6417e808-7f35-4d1a-843f-123456789012",
      "window-target-id":"e7265f13-3cc5-4f2f-97a9-123456789012",
      "status":"TIMED_OUT",
      "owner-information":"Owner"
   }
}
```

Changement d'état de la fenêtre

Les valeurs d'état valides sont `ENABLED` et `DISABLED`.

```
{
   "version":"0",
   "id":"01234567-0123-0123-0123-0123456789ab",
   "detail-type":"Maintenance Window State-change Notification",
   "source":"aws.ssm",
   "account":"123456789012",
   "time":"2016-11-16T00:58:37Z",
   "region":"us-east-2",
   "resources":[
      "arn:aws:ssm:us-east-2:123456789012:maintenancewindow/mw-123456789012345678"
   ],
   "detail":{
      "window-id":"mw-123456789012",
      "status":"DISABLED"
   }
}
```

AWS Systems Manager Evénements Parameter Store

Vous trouverez ci-dessous des exemples d'événements pour Systems Manager Parameter Store.

Création de paramètre


```
{
  "version": "0",
  "id": "6a7e4feb-b491-4cf7-a9f1-bf3703497718",
  "detail-type": "Parameter Store Change",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2017-05-22T16:43:48Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:ssm:us-east-2:123456789012:parameter/MyExampleParameter"
  ],
  "detail": {
    "operation": "Create",
    "name": "MyExampleParameter",
    "type": "String",
    "description": "Sample Parameter"
  }
}
```


Mise à jour de paramètre


```
{
  "version": "0",
  "id": "9547ef2d-3b7e-4057-b6cb-5fdf09ee7c8f",
  "detail-type": "Parameter Store Change",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2017-05-22T16:44:48Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:ssm:us-east-2:123456789012:parameter/MyExampleParameter"
  ],
  "detail": {
    "operation": "Update",
    "name": "MyExampleParameter",
    "type": "String",
```

```
    "description": "Sample Parameter"
  }
}
```

Suppression de paramètre

```
{
  "version": "0",
  "id": "80e9b391-6a9b-413c-839a-453b528053af",
  "detail-type": "Parameter Store Change",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2017-05-22T16:45:48Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:ssm:us-east-2:123456789012:parameter/MyExampleParameter"
  ],
  "detail": {
    "operation": "Delete",
    "name": "MyExampleParameter",
    "type": "String",
    "description": "Sample Parameter"
  }
}
```

AWS Systems Manager EvénementsOpsCenter

Créer des notifications OpsCenter OpsItem

```
{
  "version": "0",
  "id": "aae66adc-7aac-f0c0-7854-7691e8c079b8",
  "detail-type": "OpsItem Create",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2023-10-19T02:48:11Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ssm:us-west-2:123456789012:opsitem/oi-123456abcdef"
  ],
  "detail": {
    "created-by": "arn:aws:iam::123456789012:user/JohnDoe",
```

```
    "created-time": "2023-10-19T02:46:53.629361Z",
    "source": "aws.ssm",
    "status": "Open",
    "ops-item-id": "oi-123456abcdef",
    "title": "An issue title",
    "ops-item-type": "/aws/issue",
    "description": "A long description may appear here"
  }
}
```

Mettre à jour une notification OpsCenter OpsItem

```
{
  "version": "0",
  "id": "2fb5b168-b725-41dd-a890-29311200089c",
  "detail-type": "OpsItem Update",
  "source": "aws.ssm",
  "account": "123456789012",
  "time": "2023-10-19T02:48:11Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ssm:us-west-2:123456789012:opsitem/oi-123456abcdef"
  ],
  "detail": {
    "created-by": "arn:aws:iam::123456789012:user/JohnDoe",
    "created-time": "2023-10-19T02:46:54.049271Z",
    "modified-by": "arn:aws:iam::123456789012:user/JohnDoe",
    "modified-time": "2023-10-19T02:46:54.337354Z",
    "source": "aws.ssm",
    "status": "Open",
    "ops-item-id": "oi-123456abcdef",
    "title": "An issue title",
    "ops-item-type": "/aws/issue",
    "description": "A long description may appear here"
  }
}
```

AWS Systems Manager EvénementsRun Command

Notification de changement de statut Run Command

```
{
    "version": "0",
```

```
    "id": "51c0891d-0e34-45b1-83d6-95db273d1602",
    "detail-type": "EC2 Command Status-change Notification",
    "source": "aws.ssm",
    "account": "123456789012",
    "time": "2016-07-10T21:51:32Z",
    "region": "us-east-2",
    "resources": ["arn:aws:ec2:us-east-2:123456789012:instance/i-abcd1111"],
    "detail": {
        "command-id": "e8d3c0e4-71f7-4491-898f-c9b35bee5f3b",
        "document-name": "AWS-RunPowerShellScript",
        "expire-after": "2016-07-14T22:01:30.049Z",
        "parameters": {
            "executionTimeout": ["3600"],
            "commands": ["date"]
        },
        "requested-date-time": "2016-07-10T21:51:30.049Z",
        "status": "Success"
    }
}
```

Notification de changement de statut d'une invocation Run Command

```
{
    "version": "0",
    "id": "4780e1b8-f56b-4de5-95f2-95db273d1602",
    "detail-type": "EC2 Command Invocation Status-change Notification",
    "source": "aws.ssm",
    "account": "123456789012",
    "time": "2016-07-10T21:51:32Z",
    "region": "us-east-2",
    "resources": ["arn:aws:ec2:us-east-2:123456789012:instance/i-abcd1111"],
    "detail": {
        "command-id": "e8d3c0e4-71f7-4491-898f-c9b35bee5f3b",
        "document-name": "AWS-RunPowerShellScript",
        "instance-id": "i-9bb89e2b",
        "requested-date-time": "2016-07-10T21:51:30.049Z",
        "status": "Success"
    }
}
```

AWS Systems Manager EvénementsState Manager

Changement d'état d'association State Manager

```
{
   "version":"0",
   "id":"db839caf-6f6c-40af-9a48-25b2ae2b7774",
   "detail-type":"EC2 State Manager Association State Change",
   "source":"aws.ssm",
   "account":"123456789012",
   "time":"2017-05-16T23:01:10Z",
   "region":"us-east-2",
   "resources":[
      "arn:aws:ssm:us-east-2::document/AWS-RunPowerShellScript"
   ],
   "detail":{
      "association-id":"6e37940a-23ba-4ab0-9b96-5d0a1a05464f",
      "document-name":"AWS-RunPowerShellScript",
      "association-version":"1",
      "document-version":"Optional.empty",
      "targets":"[{\"key\":\"InstanceIds\",\"values\":[\"i-12345678\"]}]",
      "creation-date":"2017-02-13T17:22:54.458Z",
      "last-successful-execution-date":"2017-05-16T23:00:01Z",
      "last-execution-date":"2017-05-16T23:00:01Z",
      "last-updated-date":"2017-02-13T17:22:54.458Z",
      "status":"Success",
      "association-status-aggregated-count":"{\"Success\":1}",
      "schedule-expression":"cron(0 */30 * * * ? *)",
      "association-cwe-version":"1.0"
   }
}
```

Changements d'état d'association d'instance State Manager

```
{
   "version":"0",
   "id":"6a7e8feb-b491-4cf7-a9f1-bf3703467718",
   "detail-type":"EC2 State Manager Instance Association State Change",
   "source":"aws.ssm",
   "account":"123456789012",
   "time":"2017-02-23T15:23:48Z",
   "region":"us-east-2",
   "resources":[
      "arn:aws:ec2:us-east-2:123456789012:instance/i-12345678",
      "arn:aws:ssm:us-east-2:123456789012:document/my-custom-document"
   ],
   "detail":{
```

```
      "association-id":"34fcb7e0-9a14-4984-9989-0e04e3f60bd8",
      "instance-id":"i-12345678",
      "document-name":"my-custom-document",
      "document-version":"1",
      "targets":"[{\"key\":\"instanceids\",\"values\":[\"i-12345678\"]}]",
      "creation-date":"2017-02-23T15:23:48Z",
      "last-successful-execution-date":"2017-02-23T16:23:48Z",
      "last-execution-date":"2017-02-23T16:23:48Z",
      "status":"Success",
      "detailed-status":"",
      "error-code":"testErrorCode",
      "execution-summary":"testExecutionSummary",
      "output-url":"sampleurl",
      "instance-association-cwe-version":"1"
   }
}
```


Exemples de scénarios : cibles Systems Manager dans les règles Amazon EventBridge

Lorsque vous spécifiez la cible à appeler dans une règle Amazon EventBridge, vous pouvez choisir parmi plus de 20 types de cible et ajouter jusqu'à cinq cibles à chaque règle.

Vous pouvez ainsi choisir différentes cibles, dont Automation, OpsCenter et Run Command, qui sont des fonctionnalités de AWS Systems Manager, en tant qu'actions cibles lorsqu'un événement EventBridge se produit.

Voici plusieurs exemples d'utilisation de ces fonctionnalités comme cible d'une règle EventBridge.

Exemples Automation

Vous pouvez configurer une règle EventBridge pour démarrer des flux de travail Automation lorsque des événements semblables à ceux qui suivent se produisent :

- Lorsqu'une alarme Amazon CloudWatch signale qu'un nœud géré a échoué à une vérification d'état (`StatusCheckFailed_Instance=1`), exécutez le runbook Automation `AWSSupport-ExecuteEC2Rescue` sur ce nœud.
- Lorsqu'un événement `EC2 Instance State-change Notification` se produit du fait de l'exécution d'une nouvelle instance Amazon Elastic Compute Cloud (Amazon EC2), exécutez le runbook Automation `AWS-AttachEBSVolume` sur l'instance.

- Lorsqu'un volume Amazon Elastic Block Store (Amazon EBS) est créé et disponible, exécutez le runbook Automation `AWS-CreateSnapshot` sur le volume.

Exemples OpsCenter

Vous pouvez configurer une règle EventBridge pour créer un nouvel OpsItemlorsque des incidents semblables à ceux qui suivent se produisent :

- Un événement de limitation pour Amazon DynamoDB se produit ou les performances du volume Amazon EBS se sont dégradées.
- Un groupe Amazon EC2 Auto Scaling ne parvient pas à lancer un nœud ou un flux de travail Systems Manager Automation échoue.
- Une instance EC2 passe d'un état `Running` à un état `Stopped`.

Exemples Run Command

Vous pouvez configurer une règle EventBridge pour exécuter un document de commande Systems Manager dans Run Command lorsque des événements semblables à ceux qui suivent se produisent :

- Lorsqu'un groupe Auto Scaling est sur le point de se terminer, un script Run Command peut capturer les fichiers journaux du nœud avant qu'il arrive à son terme.
- Lorsqu'un nouveau nœud est créé dans un groupe Auto Scaling, une action cible Run Command peut activer le rôle de serveur Web ou installer des logiciels sur ce nœud.
- Lorsqu'un nœud géré est jugé non conforme, une action cible Run Command peut mettre à jour les correctifs sur ce nœud en exécutant le document `AWS-RunPatchBaseline`.

Surveillance des changements d'état du Systems Manager à l'aide des notifications Amazon SNS

> **Note**
>
> Les rubriques FIFO d'Amazon Simple Notification Service ne sont pas prises en charge.

Vous pouvez configurer Amazon Simple Notification Service (Amazon SNS) de sorte à envoyer des notifications sur le statut des commandes que vous envoyez à l'aide de Run Command ou Maintenance Windows, qui sont des fonctionnalités de AWS Systems Manager. Amazon SNS coordonne et gère la réception ou l'envoi de notifications aux points de terminaison ou aux clients abonnés aux 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 nœuds, vous pouvez recevoir une notification pour chaque copie de la commande envoyée à un nœud 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 qu'Amazon SNS ? dans le Guide du développeur Amazon Simple Notification Service. Pour voir des exemples de la structure des données JSON incluses dans la notification Amazon SNS fournie par Maintenance Windows et Run Command, consultez Exemple de notifications Amazon SNS pour AWS Systems Manager.

Configurer les notifications Amazon SNS pour AWS Systems Manager

Run Command et les tâches Maintenance Windows 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 :

- En cours
- Réussite
- Échec
- Expiré
- Annulé

Pour de plus amples informations sur les conditions qui entraînent ces changements de statut, consultez la section Comprendre les états des commandes.

> **Note**
>
> Les commandes envoyées à l'aide de la fonctionnalité Run Command signalent également les statuts Annulation en cours et En suspens. Ces statuts ne sont pas capturés par les notifications Amazon SNS.

Résumé de commandes associé aux notifications Amazon SNS

Si vous configurez Run Command ou une tâche Run Command dans votre fenêtre de maintenance pour les notifications Amazon SNS, Amazon SNS envoie des messages récapitulatifs qui incluent les informations suivantes.

Champ	Type	Description
eventTime	Chaîne	Heure à laquelle l'événement a été initié. L'horodatage est important, car Amazon SNS ne garantit pas l'ordre de transmission des messages. Exemple : 2016-04-26T13:15:30Z
documentName	Chaîne	Nom du document SSM utilisé pour exécuter cette commande.
commandId	Chaîne	ID généré par la fonctionnalité Run Command 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 démarré, l'exécution n'a pas lieu.
outputS3BucketName	Chaîne	Le compartiment Amazon Simple Storage Service (Amazon S3) dans lequel les réponses à l'exécution de la commande doivent être stockées.
outputS3KeyPrefix	Chaîne	Chemin du répertoire Amazon S3 à l'intérieur du compartim

Champ	Type	Description
		ent dans lequel les réponses à l'exécution de la commande doivent être stockées.
requestedDateTime	Chaîne	Heure et date auxquelles la demande a été envoyée à ce nœud spécifique.
instanceIds	StringList	Nœuds qui étaient ciblés par la commande. **Note** Les ID d'instance ne sont inclus dans le message récapitul atif que si la tâche Run Command ciblait directement les ID d'instance. Les ID d'instance ne sont pas inclus dans le message récapitul atif si la tâche Run Command a été émise en utilisant un ciblage en fonction des balises.
status	Chaîne	Statut de la commande.

Notifications Amazon SNS basées sur les appels

Si vous envoyez une commande à plusieurs nœuds, Amazon SNS peut envoyer des messages concernant chaque copie ou appel de la commande. Les messages incluent les informations suivantes.

Champ	Type	Description
eventTime	Chaîne	Heure à laquelle l'événement a été initié. L'horodatage est important, car Amazon SNS ne garantit pas l'ordre de transmission des messages. Exemple : 2016-04-26T13:15:30Z
documentName	Chaîne	Nom du document Systems Manager (document SSM) utilisé pour exécuter cette commande.
requestedDateTime	Chaîne	Heure et date auxquelles la demande a été envoyée à ce nœud spécifique.
commandId	Chaîne	ID généré par la fonctionnalité Run Command une fois que la commande a été envoyée.
instanceId	Chaîne	Instance qui était ciblée par la commande.
status	Chaîne	Statut de la commande pour cet appel.

Pour configurer les notifications Amazon SNS en cas de changement de statut d'une commande, effectuez les tâches suivantes.

> **Note**
>
> Si vous ne configurez pas les notifications Amazon SNS pour votre fenêtre de maintenance, vous pouvez ignorer la tâche 5 décrite plus loin dans cette rubrique.

Rubriques

Tâche 1 : Créer une rubrique Amazon SNS et s'y abonner

Une rubrique Amazon SNS est un canal de communication utilisé par des tâches Run Command et Run Command, qui sont enregistrées auprès d'une fenêtre de maintenance, pour envoyer des notifications relatives au statut de vos commandes. Amazon SNS prend en charge différents protocoles de communication, notamment HTTP/S, les e-mails et d'autres Services AWS comme Amazon Simple Queue Service (Amazon SQS). Pour commencer, nous vous recommandons de commencer par le protocole de messagerie. Pour obtenir des informations sur la création d'une rubrique, consultez Création d'une rubrique Amazon SNS dans le Guide du développeur Amazon Simple Notification Service.

> **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 plus d'informations sur l'abonnement à une rubrique, consultez Abonnement à une rubrique Amazon SNS dans le Guide du développeur Amazon Simple Notification Service.

Amazon SNS envoie un e-mail de confirmation à partir d'AWS Notifications à l'adresse e-mail que vous spécifiez. Ouvrez cet e-mail et sélectionnez le lien Confirm subscription (Confirmer l'abonnement).

Vous recevrez un message d'accusé de réception d'AWS. Amazon SNS est maintenant configuré pour recevoir des notifications et envoyer la notification par e-mail à l'adresse spécifiée.

Tâche 2 : Créer une politique IAM pour les notifications Amazon SNS

Utilisez la procédure suivante afin de créer une politique AWS Identity and Access Management (IAM) personnalisée qui fournit les autorisations nécessaires pour lancer des notifications Amazon SNS.

Pour créer une politique IAM personnalisée 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 Policies, puis Create Policy. (Si un bouton Get Started (Mise en route) est affiché, sélectionnez-le, puis sélectionnez Create Policy [Créer une politique].)
3. Sélectionnez l'onglet JSON.
4. Remplacez le contenu par défaut par la commande suivante :

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "sns:Publish"
            ],
            "Resource": "arn:aws:sns:region:account-id:sns-topic-name"
        }
    ]
}
```

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

account-id représente l'identifiant à 12 chiffres de votre Compte AWS au format 123456789012.

sns-topic-name représente le nom de la rubrique Amazon SNS que vous souhaitez utiliser pour publier des notifications.

5. Sélectionnez Suivant : Étiquettes.
6. (Facultatif) Ajoutez une ou plusieurs paires clé-valeur d'identification afin d'organiser, de suivre ou de contrôler l'accès pour cette politique.
7. Choisissez Next: Review (Suivant : Vérification).
8. Sur la page Examiner une politique, dans le champ Nom, saisissez un nom pour la politique en ligne. Par exemple : **my-sns-publish-permissions**.
9. (Facultatif) Dans le champ Description, saisissez une description pour la politique.
10. Sélectionnez Créer une politique.

Tâche 3 : Créer un rôle IAM pour les notifications Amazon SNS

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 initier des notifications Amazon SNS. Dans toutes les procédures suivantes, ce rôle est appelé rôle IAM Amazon SNS.

Pour créer un rôle de service IAM pour les notifications Amazon SNS

1. Connectez-vous à l'outil AWS Management Console, puis ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation de la console IAM, sélectionnez Roles (Rôles), puis Create role (Créer un rôle).
3. Choisissez le type de rôle du Service AWS, puis choisissez Systems Manager.
4. Choisissez le cas d'utilisation de Systems Manager. Ensuite, choisissez Next (Suivant).
5. Sur la page Attacher des politiques d'autorisations, cochez la case située à gauche du nom de la politique personnalisée que vous avez créée à la tâche 2. Par exemple : **my-sns-publish-permissions**.
6. (Facultatif) Définissez une limite d'autorisations. Il s'agit d'une fonctionnalité avancée disponible pour les rôles de service, mais pas les rôles liés à un service.

Développez la section Permissions boundary (Limite d'autorisations) et sélectionnez Use a permissions boundary to control the maximum role permissions (Utiliser une limite d'autorisations pour contrôler le nombre maximum d'autorisations de rôle). IAM inclut une liste des politiques gérées par AWS et des politiques gérées par le client dans votre compte. Sélectionnez la politique à utiliser pour la limite d'autorisations ou choisissez Créer une politique pour ouvrir un nouvel onglet de navigateur et créer une nouvelle politique de bout en bout. Pour plus d'informations, consultez Création de politiques IAM dans le Guide de l'utilisateur IAM. Une fois la politique créée, fermez cet onglet et revenez à l'onglet initial pour sélectionner la politique à utiliser pour la limite d'autorisations.

7. Choisissez Next (Suivant).
8. Si possible, saisissez un nom de rôle ou le suffixe d'un nom de rôle vous permettant d'identifier l'objectif du rôle. Les noms de rôle doivent être uniques dans votre Compte AWS. Ils ne sont pas sensibles à la casse. Par exemple, vous ne pouvez pas créer deux rôles nommés **`PRODROLE`** et **`prodrole`**. Différentes entités peuvent référencer le rôle et il n'est donc pas possible de modifier son nom après sa création.
9. (Facultatif) Pour Description, saisissez une description pour le nouveau rôle.
10. Choisissez Edit (Modifier) dans les sections Step 1: Select trusted entities (Étape 1 : sélection d'entités de confiance) ou Step 2: Select permissions (Étape 2 : sélection d'autorisations) pour modifier les cas d'utilisation et les autorisations pour le rôle.
11. (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 la rubrique Balisage des ressources IAM dans le Guide de l'utilisateur IAM.
12. Passez en revue les informations du rôle, puis choisissez Créer un rôle.
13. Choisissez le nom du rôle, puis copiez ou notez la valeur de Role ARN (ARN du rôle). Cet Amazon Resource Name (ARN) pour le rôle est utilisé lorsque vous envoyez une commande configurée pour renvoyer des notifications Amazon SNS.
14. La page Summary (Récapitulatif) s'ouvre.

Tâche 4 : Configuration de l'accès utilisateur

Si des autorisations d'administrateur sont attribuées à une entité IAM (utilisateur, rôle ou groupe), l'utilisateur ou le rôle a accès à Run Command et Maintenance Windows, des fonctionnalités AWS Systems Manager.

Pour les entités sans autorisations d'administrateur, un administrateur doit accorder les autorisations suivantes à l'entité IAM :

- La politique gérée `AmazonSSMFullAccess`, ou une politique qui fournit des autorisations comparables.
- Les autorisations `iam:PassRole` pour le rôle créé dans Tâche 3 : Créer un rôle IAM pour les notifications Amazon SNS. Par exemple :

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::account-id:role/sns-role-name"
        }
    ]
}
```

Pour activer l'accès, ajoutez des autorisations à vos utilisateurs, groupes ou rôles :

- Utilisateurs et groupes dans AWS IAM Identity Center :

  Créez un jeu d'autorisations. Suivez les instructions de la rubrique Création d'un jeu d'autorisations du Guide de l'utilisateur AWS IAM Identity Center.
- Utilisateurs gérés dans IAM par un fournisseur d'identité :

  Créez un rôle pour la fédération d'identité. Pour plus d'informations, voir la rubrique Création d'un rôle pour un fournisseur d'identité tiers (fédération) du Guide de l'utilisateur IAM.
- Utilisateurs IAM :
  - Créez un rôle que votre utilisateur peut assumer. Suivez les instructions de la rubrique Création d'un rôle pour un utilisateur IAM du Guide de l'utilisateur IAM.
  - (Non recommandé) Attachez une politique directement à un utilisateur ou ajoutez un utilisateur à un groupe d'utilisateurs. Suivez les instructions de la rubrique Ajout d'autorisations à un utilisateur (console) du Guide de l'utilisateur IAM.

Pour configurer l'accès utilisateur et associer la politique **`iam:PassRole`** à un compte utilisateur

1. Dans le panneau de navigation IAM, sélectionnez Users (Utilisateurs), puis le compte utilisateur à configurer.
2. Sous l'onglet Permissions (Autorisations), dans la liste des politiques, vérifiez que la politique **`AmazonSSMFullAccess`** est répertoriée ou qu'une politique comparable autorise le compte à accéder à Systems Manager.
3. Sélectionnez Ajouter une politique en ligne.
4. Dans la page Créer une politique, sélectionnez l'onglet Éditeur visuel.
5. Sélectionnez Choose a service (Sélectionner un service), puis IAM.
6. Pour Actions (Actions), dans la zone de texte Filter actions (Filtrer les actions), saisissez **`PassRole`**, puis sélectionnez la case à cocher en regard de PassRole.
7. Pour Resources (Ressources), vérifiez que Specific (Spécifique) est sélectionné, puis sélectionnez Add ARN (Ajouter un ARN).
8. Dans le champ Specify ARN for role (Spécifier l'ARN du rôle), collez l'ARN du rôle IAM 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).
9. Choisissez Add (Ajouter).
10. Sélectionnez Review policy (Examiner une politique).
11. Sur la page Review Policy (Examiner une politique), saisissez un nom, puis choisissez Create Policy (Créer une politique).

Tâche 5 : Attachement de la politique iam:PassRole à votre rôle de fenêtre de maintenance

Lorsque vous enregistrez une tâche Run Command 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 Run Command enregistrée, attachez une politique `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, vous pouvez ignorer cette tâche.

La politique `iam:PassRole` autorise le rôle de service Maintenance Windows à transmettre le rôle IAM Amazon SNS créé au cours de la tâche 3 au service Amazon SNS. La procédure suivante montre comment attacher la politique `iam:PassRole` au rôle de service Maintenance Windows.

> **Note**
>
> Utilisez un rôle de service personnalisé pour votre fenêtre de maintenance afin d'envoyer des notifications concernant les tâches Run Command enregistrées. Pour plus d'informations, consultez Configuration de Maintenance Windows.
> Si vous devez créer une fonction du service personnalisée pour les tâches de la fenêtre de maintenance, consultez la rubrique Utiliser la console pour configurer les autorisations pour les fenêtres de maintenance.

Pour attacher la politique **`iam:PassRole`** au rôle Maintenance Windows

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation, sélectionnez Roles (Rôles), puis sélectionnez le rôle IAM Amazon SNS créé au cours de la tâche 3.
3. Copiez le Role ARN (ARN de rôle) ou notez-le et revenez à la section Roles (Rôles) de la console IAM.
4. Sélectionnez le rôle de service Maintenance Windows personnalisé que vous avez créé depuis la liste Role name (Nom de rôle).
5. Sous l'onglet Permissions (Autorisations), vérifiez que la politique `AmazonSSMMaintenanceWindowRole` est répertoriée ou qu'une politique comparable accorde à la fenêtre de maintenance l'autorisation d'accéder à l'API Systems Manager. Si ce n'est pas le cas, choisissez Attacher des politiques pour l'attacher.
6. Choisissez Add permissions, Create inline policy (Ajouter des autorisations, Créer une politique en ligne).
7. Sélectionnez l'onglet Visual Editor.
8. Pour Service, sélectionnez IAM.
9. Pour Actions (Actions), dans la zone de texte Filter actions (Filtrer les actions), saisissez **`PassRole`**, puis sélectionnez la case à cocher en regard de PassRole.
10. Pour Ressources, sélectionnez Spécifique, puis Ajouter un ARN.

11. Dans la zone Specify ARN for role (Spécifier l'ARN pour le rôle), collez l'ARN du rôle IAM Amazon SNS créé dans la tâche 3, puis sélectionnez Add (Ajouter).
12. Sélectionnez Review policy (Examiner une politique).
13. Sur la page Examiner une politique indiquez un nom pour la politique `PassRole`, puis sélectionnez Créer une politique.

Exemple de notifications Amazon SNS pour AWS Systems Manager

Vous pouvez configurer Amazon Simple Notification Service (Amazon SNS) de sorte à envoyer des notifications sur le statut des commandes que vous envoyez à l'aide de Run Command ou Maintenance Windows, qui sont des fonctionnalités de AWS Systems Manager.

> **Note**
>
> Ce guide ne traite pas de la configuration des notifications pour la fonctionnalité Run Command ou une Maintenance Windows. Pour de plus amples informations sur la configuration de la fonctionnalité Run Command ou d'une Maintenance Windows pour envoyer des notifications Amazon SNS sur le statut des commandes, veuillez consulter Configurer les notifications Amazon SNS pour AWS Systems Manager.

Les exemples suivants montrent la structure de la sortie JSON renvoyée par les notifications Amazon SNS configurées pour Run Command ou Maintenance Windows.

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 Run Command pour envoyer une commande qui renvoie des notifications de statut

Les procédures suivantes montrent comment utiliser le AWS Command Line Interface (AWS CLI) ou la AWS Systems Manager console pour envoyer une commandeRun Command, une fonctionnalité de AWS Systems Manager, configurée pour renvoyer des notifications d'état.

Envoi d'une tâche Run Command qui renvoie des notifications (console)

Utilisez la procédure suivante pour envoyer via la fonctionnalité Run Command 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 AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Run Command.
3. Sélectionnez Run Command (Exécuter la commande).
4. Dans la liste Command document (Document de commande), sélectionnez 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 Targets (Cibles), sélectionnez les nœuds gérés sur lesquels vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant des instances ou des appareils de périphérie manuellement ou en spécifiant un groupe de ressources.

   > **Tip**
   >
   > Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés 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. Pour Rate control (Contrôle de débit) :

   - Dans Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage de nœuds gérés sur lesquels exécuter simultanément la commande.

     > **Note**
     >
     > Si vous avez sélectionné des cibles en spécifiant les balises appliquées aux nœuds gérés ou en spécifiant des groupes de ressources AWS , et que vous n'êtes pas certain du nombre de nœuds gérés ciblés, limitez le nombre de cibles autorisées à exécuter simultanément le document en indiquant un pourcentage.

- Dans Error threshold (Seuil d'erreur), indiquez quand arrêter l'exécution de la commande sur les autres nœuds gérés après l'échec de celle-ci sur un certain nombre ou un certain pourcentage de nœuds. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les nœuds gérés sur lesquels la commande est toujours 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). Saisissez 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 (pour les instances EC2) ou de la fonction du service IAM (pour les machines activées par un système hybride) attribués à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, assurez-vous que le profil d'instance ou le rôle de service IAM associé au nœud géré 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 la section IAM Role (Rôle IAM), choisissez l'ARN du rôle IAM Amazon SNS que vous avez créé au cours de la tâche 3 dans Surveillance des changements d'état du Systems Manager à l'aide des notifications Amazon SNS.
12. Dans le champ SNS topic (Rubrique SNS), saisissez l'ARN de la rubrique Amazon SNS à utiliser.
13. Pour Event notifications (Notifications d'événement), sélectionnez les événements pour lesquels vous souhaitez recevoir des notifications.
14. Pour Change notifications (Notification de modification), choisissez de recevoir des notifications uniquement pour le résumé des commandes (Command status changes (Changements d'état des commandes)) ou pour chaque copie d'une commande envoyée à plusieurs nœuds (Command status on each instance changes (L'état des commandes sur chaque instance change)).

15. Cliquez sur Exécuter.
16. Recherchez un message d'Amazon SNS dans votre messagerie et ouvrez l'e-mail. L'envoi de l'e-mail peut prendre plusieurs minutes à Amazon SNS.

Envoi d'une tâche Run Command qui renvoie des notifications (interface de ligne de commande)

Utilisez la procédure suivante pour envoyer via la fonctionnalité Run Command 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 le AWS CLI.
2. Spécifiez les paramètres de la commande suivante pour cibler les nœuds en fonction des ID de nœud géré.

```
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"}'
```

Voici un 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"}'
```

Voici un 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 Entrée.
4. Recherchez un message d'Amazon SNS dans votre messagerie et ouvrez l'e-mail. L'envoi de l'e-mail peut prendre plusieurs minutes à Amazon SNS.

Pour plus d'informations, consultez la section send-command dans la référence des commandes AWS CLI .

Utilisation d'une fenêtre de maintenance pour envoyer une commande qui renvoie des notifications de statut

Les procédures suivantes indiquent comment enregistrer une Run Command tâche dans votre fenêtre de maintenance à l'aide de la AWS Systems Manager console ou du AWS Command Line Interface (AWS CLI). Run Commandest une capacité de AWS Systems Manager. Ces procédures expliquent aussi comment configurer la tâche Run Command pour renvoyer des notifications de statut.

Avant de commencer

Si vous n'avez pas encore créé de fenêtre de maintenance ou de cibles enregistrées, consultez Utilisation des fenêtres de maintenance (console) pour connaître les étapes permettant de créer une fenêtre de maintenance et d'enregistrer des cibles.

Pour recevoir des notifications du service Amazon Simple Notification Service (Amazon SNS), attachez une politique `iam:PassRole` au rôle de service Maintenance Windows spécifié dans la tâche enregistrée. Si vous n'avez pas ajouté d'autorisations `iam:PassRole` à votre rôle de service Maintenance Windows, consultez Tâche 5 : Attachement de la politique iam:PassRole à votre rôle de fenêtre de maintenance.

Enregistrement d'une tâche Run Command auprès d'une fenêtre de maintenance qui renvoie des notifications (console)

Utilisez la procédure suivante pour enregistrer une tâche Run Command 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 Run Command auprès de votre fenêtre de maintenance qui renvoie des notifications (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Maintenance Windows.
3. Sélectionnez la fenêtre de maintenance pour laquelle vous souhaitez enregistrer une tâche Run Command configurée pour envoyer des notifications Amazon Simple Notification Service (Amazon SNS).
4. Sélectionnez Actions, puis Register Run command task (Enregistrer une tâche Run Command).
5. (Facultatif) Dans le champ Name (Nom), saisissez un nom pour la tâche.
6. (Facultatif) Dans le champ Description, saisissez une description.
7. Dans la liste Command Document (Document de commande), sélectionnez un document de commande.
8. 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. Les tâches qui ont la même priorité sont planifiées en parallèle.
9. Dans la section Cibles, sélectionnez un groupe cible enregistré ou sélectionnez des cibles non enregistrées.
10. Pour Rate control (Contrôle de débit) :

    - Dans Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage de nœuds gérés sur lesquels exécuter simultanément la commande.

      > **Note**
      >
      > Si vous avez sélectionné des cibles en spécifiant les balises appliquées aux nœuds gérés ou en spécifiant des groupes de ressources AWS , et que vous n'êtes pas

certain du nombre de nœuds gérés ciblés, limitez le nombre de cibles autorisées à exécuter simultanément le document en indiquant un pourcentage.

- Dans Error threshold (Seuil d'erreur), indiquez quand arrêter l'exécution de la commande sur les autres nœuds gérés après l'échec de celle-ci sur un certain nombre ou un certain pourcentage de nœuds. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les nœuds gérés sur lesquels la commande est toujours en cours de traitement peuvent également envoyer des erreurs.

11. Dans la zone IAM service role (Rôle de service IAM), sélectionnez le rôle de service Maintenance Windows qui dispose des autorisations `iam:PassRole` sur le rôle SNS.

> **Note**
>
> Ajoutez des autorisations `iam:PassRole` au rôle Maintenance Windows pour permettre à Systems Manager de transmettre le rôle SNS à Amazon SNS. Si vous n'avez pas ajouté d'autorisations `iam:PassRole`, consultez Tâche 5 dans la rubrique Surveillance des changements d'état du Systems Manager à l'aide des notifications Amazon SNS.

12. (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) Saisissez les noms de compartiment et de préfixe (dossier) dans les zones.

> **Note**
>
> Les autorisations S3 qui donnent la possibilité d'écrire les données dans un compartiment S3 sont celles du profil d'instance attribué au nœud géré, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, vérifiez que le profil d'instance ou le rôle de service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

13. Dans la section Notifications SNS, procédez comme suit :

- Sélectionnez Activer les notifications SNS.

- Pour rôle IAM, sélectionnez l'Amazon Resource Name (ARN) du rôle IAM Amazon SNS que vous avez créé au cours de la tâche 3 dans Surveillance des changements d'état du Systems Manager à l'aide des notifications Amazon SNS pour initier Amazon SNS.
- Dans le champ SNS topic (Rubrique SNS), saisissez l'ARN de la rubrique Amazon SNS à utiliser.
- Pour Event type (Type d'événement), sélectionnez les événements pour lesquels vous souhaitez recevoir des notifications.
- Dans Notification type (Type de notification), choisissez de recevoir des notifications pour chaque copie d'une commande envoyée à plusieurs nœuds (appels) ou de recevoir un récapitulatif des commandes.

14. Dans la section Parameters (Paramètres), entrez les paramètres requis en fonction du document de commande que vous avez choisi.
15. Sélectionnez Register run command task (Enregistrer une tâche d'exécution de commande).
16. Après l'exécution suivante de votre fenêtre de maintenance, vérifiez vos e-mails pour savoir si vous avez reçu un message d'Amazon SNS, et ouvrez le message. Amazon SNS peut prendre quelques minutes pour envoyer le message.

Enregistrement d'une tâche Run Command auprès d'une fenêtre de maintenance qui renvoie des notifications (interface de ligne de commande)

Utilisez la procédure suivante pour enregistrer une tâche Run Command 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 Run Command 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::account-id: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:account-id:SNSTopicName",
                "NotificationEvents": [
                    "All"
                ],
                "NotificationType": "Command"
            },
            "ServiceRoleArn": "arn:aws:iam::account-id: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 plus d'informations, consultez la section register-task-with-maintenance-window dans la référence des commandes AWS CLI .

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, vérifiez vos e-mails pour savoir si vous avez reçu un message d'Amazon SNS, et ouvrez le message. Amazon SNS peut prendre quelques minutes pour envoyer le message.

Pour de plus amples informations sur l'enregistrement de tâches pour une fenêtre de maintenance à partir de la ligne de commande, consultez Enregistrer des tâches avec la fenêtre de maintenance.

Intégrations de produits et services à Systems Manager

Par défaut, AWS Systems Manager s'intègre aux Services AWS, ainsi qu'aux 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 avec Services AWS

Grâce aux documents de commande de Systems Manager (documents SSM) et aux runbooks d'automatisation, vous pouvez les utiliser AWS Systems Manager pour les intégrer à. Services AWS Pour plus d'information sur ces ressources, consultez AWS Systems Manager Documents.

Systems Manager est intégré aux éléments suivants Services AWS.

Calcul

Amazon Elastic Compute Cloud (Amazon EC2)	Amazon EC2 offre une capacité de calcul évolutive dans le AWS Cloud. L'utilisation d'Amazon EC2 vous dispense d'investir à l'avance dans du matériel et, par conséquen t, 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 les réseaux, et gérer le stockage. Systems Manager vous permet d'effectuer plusieurs tâches sur des instances EC2. Par exemple, vous pouvez lancer, configurer, gérer, maintenir et dépanner vos instances EC2, et vous y connecter en toute sécurité. Vous pouvez également utiliser Systems Manager pour déployer des logiciels, déterminer le statut

	de conformité et collecter l'inventaire de vos instances EC2. En savoir plus • Utilisation de nœuds gérés • AWS Systems Manager State Manager • AWS Systems Manager Run Command • AWS Systems Manager Patch Manager • AWS Systems Manager Session Manager • AWS Systems Manager Distributor • Conformité d'AWS Systems Manager • AWS Systems Manager Inventory
Amazon EC2 Auto Scaling	Auto Scaling vous permet de vous assurer que vous disposez du bon nombre d'instances EC2 disponibles pour gérer votre charge applicative. Vous créez des ensembles d'instances EC2, appelés groupes Auto Scaling. Systems Manager vous permet d'automatiser les procédures courantes, telles que l'applica tion de correctifs Amazon Machine Image (AMI) utilisée dans votre modèle Auto Scaling pour votre groupe Auto Scaling. En savoir plus Mise à jour d'AMIs pour des groupes Auto Scaling

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. Systems Manager vous permet de gérer des instances de conteneur à distance et d'injecter des données sensibles dans vos conteneurs, en stockant vos données sensibles dans des paramètres de Parameter Store, une fonctionnalité de Systems Manager, puis en les référençant dans votre définition de conteneur. En savoir plus • Gérer à distance des instances de conteneur à l'aide de AWS Systems Manager • Spécification de données sensibles en utilisant Systems Manager Parameter Store

AWS Lambda	Lambda est un service de calcul qui vous permet d'exécuter du code sans devoir approvisionner ou gérer des serveurs. Lambda exécute le code uniquement lorsque cela est nécessaire et se met à l'échelle automatiq uement, qu'il s'agisse de quelques requêtes par jour ou de milliers de requêtes par seconde. Systems Manager vous permet d'utiliser des fonctions Lambda dans le contenu du runbook Automation grâce à l'action `aws:invok eLambdaFunction` . Pour utiliser les paramètres des AWS Lambda fonctions Parameter Store in, vous pouvez utiliser l'extension Lambda AWS Parameter s and Secrets pour récupérer les valeurs des paramètres et les mettre en cache pour une utilisation future. En savoir plus Mettez à jour un golden AMI à l'aide de l'automatisation AWS Lambda, et Parameter Store Utilisation de paramètres de Parameter Store dans les fonctions AWS Lambda

Internet des objets (IoT)

AWS IoT Greengrass appareils principaux	AWS IoT Greengrass est un environnement d'exécution IoT Edge open source ainsi qu'un service cloud qui vous permet de créer, déployer et gérer des applications IoT sur vos appareils. Systems Manager offre un support

	natif pour les AWS IoT Greengrass principaux appareils. En savoir plus Gestion des appareils de pointe avec Systems Manager
AWS IoT appareils principaux	AWS IoTfournit les services cloud qui connectent vos appareils IoT à d'autres appareils et services AWS cloud. AWS IoT fournit un logiciel qui peut vous aider à intégrer vos appareils IoT dans des solutions AWS IoT basées sur ces appareils. Si vos appareils peuvent se connecter à AWS IoT, AWS IoT vous pouvez les connecter aux services cloud qui les AWS fournissent. Systems Manager prend en charge les appareils AWS IoT principaux tant que ces appareils sont configuré s en tant que nœuds gérés dans un environne ment hybride et multicloud. En savoir plus Utilisation de Systems Manager dans des environnements hybrides et multicloud

Stockage

Amazon Simple Storage Service (Amazon S3)	Amazon S3 est un service de stockage pour Internet. Il est conçu pour faciliter l'informa tique à l'échelle d'Internet pour les développe urs. Amazon S3 offre une interface simple de services web qui vous permet de stocker et de récupérer n'importe quelle quantité de données, à tout moment, de n'importe où sur Internet.

	Systems Manager vous permet d'exécuter des scripts distants et des documents SSM stockés dans Amazon S3. Distributor, une fonctionn alité de AWS Systems Manager, utilise Amazon S3 pour stocker des packages. Vous pouvez également envoyer une sortie à Amazon S3 pour Run Command etSession Manager, les fonctionnalités de AWS Systems Manager. En savoir plus • Exécution de scripts à partir d'Amazon S3 • Exécution de documents à partir d'emplacements distants • AWS Systems Manager Distributor • Journalisation des données de session avec Amazon S3 (console)

Outils pour développeurs

AWS CodeBuild	CodeBuildest un service de création entièrement géré dans le cloud. CodeBuild compile votre code source, exécute des tests unitaires et produit des artefacts prêts à être déployés. CodeBuild élimine le besoin de provisionner, de gérer et de dimensionner vos propres serveurs de construction. Parameter Store vous permet de stocker des informations sensibles pour vos spécifications et vos projets de création. En savoir plus • Référence de spécification de construction pour CodeBuild

	• Créez un projet de construction dans AWS CodeBuild
AWS CDK	AWS Cloud Development Kit (AWS CDK) Il s'agit d'un framework permettant de définir l'infrastructure cloud sous forme de code, avec des langages de programmation, et de la déployer via AWS CloudFormation. Application Manager vous permet de visualiser vos constructions CDK regroupées en applications, de visualiser la structure de l'application, y compris les ressources sous-jacentes, de consulter les alertes, d'étudier et de résoudre les problèmes opérationnels, et de suivre les coûts dans la console Application Manager. En savoir plus • Affichage des informations de présentation d'une application • Affichage des ressources d'application

Sécurité, identité et conformité

AWS Identity and Access Management (JE SUIS)	IAM est un service Web qui vous permet de contrôler en toute sécurité l'accès aux AWS ressources. Vous pouvez utiliser IAM pour contrôler les personnes qui s'authentifient (sont connectées) et sont autorisées (disposent d'autorisations) à utiliser des ressources. Systems Manager vous permet de contrôler l'accès aux services en utilisant IAM.

	En savoir plus • Fonctionnement d'AWS Systems Manager avec IAM • Actions, ressources et clés de condition pour AWS Systems Manager • Configurer les autorisations d'instance requises pour Systems Manager
AWS Secrets Manager	Secrets Manager facilite 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. Parameter Store vous permet de récupérer les secrets de Secrets Manager lors de l'utilisation d'autres Services AWS qui prennent déjà en charge les références aux paramètres Parameter Store. En savoir plus Référencement des secrets AWS Secrets Manager à partir des paramètres Parameter Store

AWS Security Hub	Security Hub vous offre une vue complète de vos alertes de sécurité haute priorité et votre statut de conformité sur vos Comptes AWS. Security Hub regroupe, organise et hiérarchise vos alertes de sécurité, ou résultats, parmi plusieurs. Services AWS Lorsque vous activez l'intégration entre Security Hub et Patch Manager une fonctionnalité de AWS Systems Manager Security Hub surveille l'état 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 le statut de conformité de vos correctifs et suivre d'autres résultats liés à la sécurité. Vous pouvez recevoir des alertes lorsque les nœuds de votre flotte ne sont pas conformes en matière de correctifs, et vous pouvez examiner les résultats de conformité dans la console Security Hub. Vous pouvez également intégrer Security Hub Explorer aux OpsCenter fonctionnalités de AWS Systems Manager. L'intégration à Security Hub vous permet de recevoir les résultats de Security Hub dans Explorer et OpsCenter. Les résultats de Security Hub fournissent des informations sur la sécurité, que vous pouvez utiliser dans Explorer et OpsCenter pour agréger vos problèmes de sécurité, de performance et d'exploitation dans AWS Systems Manager et prendre les mesures correspondantes adéquates.

	L'utilisation de Security Hub entraîne des frais supplémentaires. Pour de plus amples informations, consultez Tarification Security Hub. En savoir plus • Recevoir des résultats de AWS Security Hub dans Explorer • AWS Security Hub • Intégration Patch Manager avec AWS Security Hub

Chiffrement et ICP

AWS Key Management Service (AWS KMS)	AWS KMS est un service géré qui vous permet de créer de contrôler les clés de chiffrement utilisées pour chiffrer vos données. Systems Manager vous permet de AWS KMS créer des `SecureString` paramètres et de chiffrer les données de Session Manager session. En savoir plus • Comment AWS Systems ManagerParameter Store utilise-t-il AWS KMS ? • Activer le chiffrement des données de session par clé KMS (console)

Gestion et gouvernance

AWS CloudFormation	AWS CloudFormation est un service qui vous permet de modéliser et de configurer vos ressources Amazon Web Services de sorte que

	vous puissiez passer moins de temps à gérer ces ressources et consacrer plus de temps à vos applications exécutées dans AWS. Parameter Store est une source de référence s dynamiques. Les références dynamiques constituent un moyen compact et puissant de spécifier des valeurs externes qui sont stockées et gérées dans d'autres services dans vos modèles de AWS CloudFormation pile. En savoir plus Utilisation de références dynamiques pour spécifier les valeurs de modèle
AWS CloudTrail	CloudTrailest un outil Service AWS qui vous aide à autoriser la gouvernance, la conformit é, ainsi que l'audit opérationnel et des risques de votre entreprise Compte AWS. Les actions entreprises par un utilisateur, un rôle ou un Service AWS sont enregistrées sous forme d'événements dans CloudTrail. Les événement s incluent les actions entreprises dans le AWS Management Console, AWS Command Line Interface (AWS CLI), les AWS SDK et les API. Systems Manager s'intègre CloudTrail et capture la plupart des appels d'API Systems Manager sous forme d'événements. Cela inclut les appels d'API depuis la console Systems Manager et les appels vers les API Systems Manager. En savoir plus Journalisation des appels d' AWS Systems Manager API avec AWS CloudTrail

Amazon CloudWatch Logs	Amazon CloudWatch Logs vous permet de centraliser les journaux de tous vos systèmes, applications et applications Services AWS que vous utilisez. Vous pouvez ensuite les afficher, effectuer des recherches sur des codes d'erreur ou des motifs spécifiques, les filtrer en fonction de champs particuliers ou les archiver en toute sécurité pour une analyse ultérieure. Systems Manager prend en charge l'envoi de journaux pour SSM AgentRun Command, et Session Manager vers CloudWatch Logs. En savoir plus • Envoi des journaux des nœuds vers CloudWatch des journaux unifiés (CloudWatch agent) • Configuration d'Amazon CloudWatch Logs pour Run Command • Enregistrement des données de session à l'aide d'Amazon CloudWatch Logs (console)

Amazon EventBridge	EventBridgefournit un flux d'événements système en temps quasi réel qui décrit les modifications apportées aux ressources Amazon Web Services. À l'aide de règles simples que vous pouvez configurer rapidemen t, vous pouvez associer des événements et les acheminer vers une ou plusieurs fonctions ou flux cibles. EventBridge prend connaissa nce des changements opérationnels au fur et à mesure qu'ils se produisent. EventBridge répond à ces changements opérationnels et prend les mesures correctives nécessaires. Ces actions comprennent l'envoi de messages en réponse à l'environnement, l'activation de fonctions et la capture d'informations d'état. Systems Manager propose plusieurs événements pris en charge en vous EventBrid ge permettant de prendre des mesures en fonction du contenu de ces événements. En savoir plus Surveillance d'événements Systems Manager avec Amazon EventBridge **Note** Amazon EventBridge est le moyen préféré pour gérer vos événement s. CloudWatch Les événements et la même API EventBridge constitue nt le même service sous-jacent et la même API, mais EventBridge offrent davantage de fonctionnalités. Les modifications que vous apportez dans l'une CloudWatch ou l'autre console ou EventBridge sont répercutées dans

	chaque console. Pour plus d'informations, consultez le guide de EventBridge l'utilisateur Amazon.
AWS Config	AWS Configfournit une vue détaillée de la configuration des AWS ressources de votre Compte AWS. Cela comprend la façon dont les ressources sont interreliées et leur configuration. Vous pouvez ainsi suivre l'évolution des configurations et des relations dans le temps. Systems Manager est intégré et fournit plusieurs règles qui vous aident à gagner en visibilité sur vos instances EC2. AWS Config Ces règles vous permettent d'identifier les instances EC2 gérées par Systems Manager, les configurations du système d'exploitation, les mises à jour au niveau système, les applications installées, les configurations réseau, etc. En savoir plus • AWS Config types de ressources pris en charge • Enregistrement de la configuration logicielle pour des instances gérées • Affichage de l'historique d'inventaire et suivi des modifications

AWS Trusted Advisor	Trusted Advisor est un outil en ligne qui vous fournit des conseils en temps réel sur l'approvisionnement de vos ressources selon les bonnes pratiques AWS . Systems Manager héberge Trusted Advisor et vous pouvez y consulter Trusted Advisor les donnéesExplorer. En savoir plus • AWS Systems Manager Explorer • Commencer avec AWS Trusted Advisor

AWS Organizations	Organizations est un service de gestion de comptes qui vous permet de consolider plusieurs comptes au Comptes AWS sein d'une organisation que vous créez et gérez de manière centralisée. Organizations inclut des capacités de facturation consolidée et de gestion de compte vous aidant à satisfaire les besoins budgétaires, de sécurité et de conformité de votre entreprise. L'intégration entre Organizations Change Manager, une fonctionnalité de AWS Systems Manager, permet d'utiliser un compte d'administrateur délégué pour gérer les demandes de modification, les modèles de modification et les approbations pour l'ensemble de votre organisation via ce compte unique. Organisations : intégration à Inventory, une fonctionnalité de AWS Systems Manager, et vous Explorerpermet d'agréger les données d'inventaire et d'exploitation (OpsData) à partir de plusieurs Régions AWS et Comptes AWS. L'intégration entreQuick Setup, une fonctionnalité de AWS Systems Manager et Organizations automatise les tâches courantes de configuration des services et déploie des configurations de service basées sur les meilleures pratiques au sein de vos unités organisationnelles (UO).

Réseau et diffusion de contenu

AWS PrivateLink	AWS PrivateLinkvous permet de connecter en privé votre cloud privé virtuel (VPC) aux

	services de point de terminaison VPC pris en charge et aux services de point de terminaison Services AWS VPC sans avoir besoin d'une passerelle Internet, d'un périphérique NAT, d'une connexion VPN ou d'une connexion. AWS Direct Connect Systems Manager prend en charge les nœuds gérés qui se connectent aux API Systems Manager via AWS PrivateLink. Cela améliore le niveau de sécurité de vos nœuds gérés, AWS PrivateLink car tout le trafic réseau entre vos nœuds gérés, Systems Manager et Amazon EC2 est limité au réseau Amazon. Autrement dit, les nœuds gérés n'ont pas besoin d'avoir accès à Internet. En savoir plus Améliorez la sécurité des instances EC2 en utilisant des points de terminaison VPC pour Systems Manager

Analyse

Amazon Athena	Athena est un service de requêtes interactif qui vous permet d'analyser des données directe dans Amazon Simple Storage Service (Amazon S3) via la syntaxe SQL standard. En effectuan t quelques actions AWS Management Console, vous pouvez pointer Athena vers vos données stockées dans Amazon S3 et commencer à utiliser le SQL standard pour exécuter des requêtes ponctuelles et obtenir des résultats en quelques secondes.

	Systems Manager Inventory s'intègre à Athena pour vous aider à interroger les données d'inventaire provenant de plusieurs Régions AWS et. Comptes AWS Grâce à l'utilisation de la synchronisation des données de ressource s, l'intégration Athena vous permet de consulter les données d'inventaire de tous les nœuds gérés sur la page Detailed View (Vue détaillée) de la console Systems Manager Inventory. En savoir plus • Interrogation des données d'inventaire à partir de plusieurs régions et comptes • Démonstration : utiliser la synchronisation de données de ressources pour regrouper les données d'inventaire
AWS Glue	AWS Glue est un service d'extraction, de transformation et de chargement (ETL) entièrement géré qui facilite et rend abordable le classement par catégorie, le nettoyage , l'enrichissement et le déplacement des données de manière fiable entre différents flux de données et banques de données. Systems Manager analyse AWS Glue les données d'inventaire de votre compartiment S3. En savoir plus Interrogation des données d'inventaire à partir de plusieurs régions et comptes

Amazon QuickSight	Amazon QuickSight est un service d'analyse commerciale que vous pouvez utiliser pour créer des visualisations, effectuer des analyses ponctuelles et obtenir des informations commerciales à partir de vos données. Il peut identifier des sources de données AWS automatiquement et fonctionne également avec les vôtres. La synchronisation des données de ressources Systems Manager envoie les données d'inventaire collectées à partir de tous vos nœuds gérés vers un seul compartiment S3. Vous pouvez utiliser Amazon QuickSight pour interroger et analyser les données agrégées. En savoir plus • Configuration de la synchronisation de données de ressource pour Inventory • Démonstration : utiliser la synchronisation de données de ressources pour regrouper les données d'inventaire

Intégration des applications

Amazon Simple Notification Service (Amazon SNS)	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. Systems Manager génère des statuts pour plusieurs services qui peuvent être capturés par les notifications Amazon SNS.

	En savoir plus • Surveillance des changements d'état du Systems Manager à l'aide des notifications Amazon SNS • Configuration de notifications ou d'actions de déclenchement basées sur des événements Parameter Store

AWS Management Console

AWS Resource Groups	Les Resource Groups organisent vos AWS ressources. Les groupes de ressources faciliten t la gestion, le contrôle et l'automatisation des tâches simultanément sur un grand nombre de ressources. Des ressources Systems Manager telles que des nœuds gérés, des documents SSM, des fenêtres de maintenance, des paramètres Parameter Store et des référentiels de correctif s, peuvent être ajoutées à des groupes de ressources. En savoir plus Présentation de AWS Resource Groups

Rubriques

- Exécution de scripts à partir d'Amazon S3
- Référencement des secrets AWS Secrets Manager à partir des paramètres Parameter Store
- Utilisation de paramètres de Parameter Store dans les fonctions AWS Lambda

Exécution de scripts à partir d'Amazon S3

Cette section décrit le téléchargement et l'exécution de scripts à partir d'Amazon Simple Storage Service (Amazon S3). La rubrique suivante inclut des informations et une terminologie relatives à Amazon S3. Pour en savoir plus sur Amazon S3, consultez Qu'est-ce qu'Amazon S3 ? Vous pouvez exécuter différents types de scripts, notamment Ansible Playbooks, Python, Ruby, Shell et. PowerShell

Vous pouvez également télécharger un répertoire contenant plusieurs scripts. Lorsque vous exécutez le script principal dans le répertoire, tous les scripts référencés inclus dans le répertoire sont AWS Systems Manager également exécutés.

Notez les informations importantes suivantes relatives à l'exécution de scripts à partir d'Amazon S3 :

- Systems Manager ne vérifie pas que le script est à même de s'exécuter sur un nœud. Avant de télécharger et d'exécuter le script, vérifiez que le logiciel requis est installé sur le nœud. Vous pouvez également créer un document composite qui installe le logiciel par l'intermédiaire de Run Command ou de State Manager, des fonctionnalités de AWS Systems Manager, puis télécharge et exécute le script.
- Vérifiez que votre utilisateur, rôle ou groupe dispose des autorisations AWS Identity and Access Management (IAM) nécessaires pour la lecture sur le compartiment S3.
- Vérifiez que le profil d'instance de vos instances Amazon Elastic Compute Cloud (Amazon EC2) dispose bien 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, consultez Utilisation de profils d'instance dans le Guide de l'utilisateur IAM.

Exécuter des scripts Shell à partir d'Amazon S3

Les informations suivantes incluent des procédures pour vous aider à exécuter des scripts depuis Amazon Simple Storage Service (Amazon S3) à l'aide de AWS Systems Manager la console ou AWS Command Line Interface du AWS CLI(). Bien que les scripts shell soient utilisés dans les exemples, d'autres types de scripts peuvent être remplacés.

Exécuter un script Shell depuis Amazon S3 (console)

Exécuter un script Shell à partir d'Amazon S3

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Run Command.
3. Sélectionnez Run Command (Exécuter la commande).
4. In the Command document (Document de commande), sélectionnez **`AWS-RunRemoteScript`**.
5. Dans Paramètres de la commande, procédez comme suit :

   - Dans Type de source, sélectionnez S3.
   - Dans la zone de texte Source Info (Infos sur la source), saisissez les informations requises pour accéder à la source en respectant le format suivant. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations.

     > **Note**
     >
     > Remplacez https://s3.*`aws-api-domain`* par l'URL de votre compartiment. Vous pouvez copier l'URL de votre compartiment dans Amazon S3 sous l'onglet Objects (Objets).

     ```
     {"path":"https://s3.aws-api-domain/path to script"}
     ```

     Voici un exemple.

     ```
     {"path":"https://DOC-EXAMPLE-BUCKET.s3.us-west-2.amazonaws.com/scripts/shell/
     helloWorld.sh"}
     ```

   - Dans le champ Command Line (Ligne de commande), saisissez les paramètres d'exécution du script. Voici un exemple.

     ```
     helloWorld.sh argument-1 argument-2
     ```

   - (Facultatif) Dans le champ Working Directory (Répertoire de travail), saisissez le nom d'un répertoire du nœud 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 Targets (Cibles), sélectionnez les nœuds gérés sur lesquels vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant des instances ou des appareils de périphérie manuellement ou en spécifiant un groupe de ressources.

> **Tip**
>
> Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés 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. Pour Rate control (Contrôle de débit) :

   - Dans Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage de nœuds gérés sur lesquels exécuter simultanément la commande.

   > **Note**
   >
   > Si vous avez sélectionné des cibles en spécifiant les balises appliquées aux nœuds gérés ou en spécifiant des groupes de ressources AWS , et que vous n'êtes pas certain du nombre de nœuds gérés ciblés, limitez le nombre de cibles autorisées à exécuter simultanément le document en indiquant un pourcentage.

   - Dans Error threshold (Seuil d'erreur), indiquez quand arrêter l'exécution de la commande sur les autres nœuds gérés après l'échec de celle-ci sur un certain nombre ou un certain pourcentage de nœuds. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les nœuds gérés sur lesquels la commande est toujours 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). Saisissez 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 (pour les instances EC2) ou de la fonction du service IAM (pour les machines activées par un système hybride) attribués à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, assurez-vous que le profil d'instance ou le rôle de service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

10. Dans la section SNS notifications (Notifications SNS), si vous souhaitez envoyer des notifications sur le statut d'exécution des commandes, cochez la case Enable SNS notifications (Activer les notifications SNS).

    Pour plus d'informations sur la configuration des notifications Amazon SNS pour Run Command, consultez Surveillance des changements d'état du Systems Manager à l'aide des notifications Amazon SNS.

11. Cliquez sur Exécuter.

Exécuter un script Shell depuis Amazon S3 (ligne de commande)

1. Installez et configurez le AWS Command Line Interface (AWS CLI), si ce n'est pas déjà fait.

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l' AWS CLI.

2. Exécutez la commande suivante. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations.

   > **Note**
   >
   > Remplacez https://s3.*`aws-api-domain`* par l'URL de votre compartiment. Vous pouvez copier l'URL de votre compartiment dans Amazon S3 sous l'onglet Objects (Objets).

Linux & macOS

```
aws ssm send-command \
    --document-name "AWS-RunRemoteScript" \
    --output-s3-bucket-name "bucket-name" \
    --output-s3-key-prefix "key-prefix" \
    --targets "Key=InstanceIds,Values=instance-id" \
    --parameters '{"sourceType":["S3"],"sourceInfo":["{\"path\":\"https://
s3.aws-api-domain/script path\"}"],"commandLine":["script name and arguments"]}'
```

Windows

```
aws ssm send-command ^
    --document-name "AWS-RunRemoteScript" ^
    --output-s3-bucket-name "bucket-name" ^
    --output-s3-key-prefix "key-prefix" ^
    --targets "Key=InstanceIds,Values=instance-id" ^
    --parameters "sourceType"="S3",sourceInfo='{\"path\":\"https://s3.aws-api-
domain/script path\"}',"commandLine"="script name and arguments"
```

PowerShell

```
Send-SSMCommand `
    -DocumentName "AWS-RunRemoteScript" `
    -OutputS3BucketName "bucket-name" `
    -OutputS3KeyPrefix "key-prefix" `
    -Target @{Key="InstanceIds";Values=@("instance-id")} `
    -Parameter @{ sourceType="S3";sourceInfo='{"path": "https://s3.aws-api-
domain/script path"}',; "commandLine"="script name and arguments"}
```

Référencement des secrets AWS Secrets Manager à partir des paramètres Parameter Store

AWS Secrets Manager vous 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, une des fonctionnalités de AWS Systems Manager, est désormais intégré à Secrets Manager afin que vous puissiez récupérer les secrets de secrets Manager lors de l'utilisation d'autres Services AWS qui prennent déjà en charge les références aux paramètres Parameter Store. Ces

services comprennent Amazon Elastic Compute Cloud (Amazon EC2), Amazon Elastic Container Service (Amazon ECS), AWS Lambda, AWS CloudFormation, AWS CodeBuild, AWS CodeDeploy et d'autres fonctionnalités de 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 la création et la gestion de Secrets Manager, consultez Qu'est-ce qu'AWS Secrets Manager ? dans le Guide de l'utilisateur AWS Secrets Manager.

Restrictions

Notez les restrictions suivantes lorsque vous utilisez Parameter Store pour référencer des secrets Secrets Manager :

- Vous ne pouvez récupérer des secrets Secrets Manager qu'en utilisant les opérations d'API GetParameter et GetParameters. Les opérations de modification et les opérations d'API d'interrogation avancée, telles que DescribeParameters ou GetParametersByPath, ne sont pas prises en charge pour Secrets Manager.
- Vous pouvez utiliser l'AWS Command Line Interface (AWS CLI), les AWS Tools for Windows PowerShell et les kits SDK pour récupérer un secret à l'aide de Parameter Store.
- Lorsque vous récupérez un secret Secrets Manager à partir de Parameter Store, le nom doit commencer par le chemin d'accès réservé suivant : /aws/reference/secretsmanager/*secret-_ID*.

  Voici un exemple: `/aws/reference/secretsmanager/CFCreds1`
- Parameter Store honore les politiques AWS Identity and Access Management (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 honore les étapes de version de Secrets Manager. Si vous référencez une étape de version, celle-ci utilise 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 en utilisant Parameter Store

La procédure suivante explique comment référencer un secret Secrets Manager à l'aide des API Parameter Store. 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 d'autorisations administrateur dans Secrets Manager et Systems Manager, vous pouvez référencer ou extraire des secrets en utilisant des API Parameter Store. Si vous référencez un secret Secrets Manager dans un paramètre Parameter Store 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.

> **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.

Pour référencer un secret Secrets Manager en utilisant Parameter Store

1. Créez un secret dans Secrets Manager. Pour plus d'informations, veuillez consulter Création et gestion des secrets avec AWS Secrets Manager.
2. Référencez un secret à l'aide de l'AWS CLI, des AWS Tools for Windows PowerShell ou du kit SDK. Lorsque vous référencez un secret Secrets Manager, le nom doit commencer par le chemin d'accès réservé suivant : `/aws/reference/secretsmanager/`. En spécifiant ce chemin, Secrets Manager sait comment récupérer le secret à partir de Secrets Manager au lieu de Parameter Store. Voici quelques exemples de noms qui référencent correctement les secrets Secrets Manager, `CFCreds1` et `DBPass`, en utilisant Parameter Store.

   - `/aws/reference/secretsmanager/CFCreds1`
   - `/aws/reference/secretsmanager/DBPass`

Voici un exemple de code Java qui référence une clé d'accès et 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. Bien que les données de configuration et les informations d'identification soient stockées dans des services distincts, les deux ensembles de données sont accessibles à partir de Parameter Store à l'aide de l'API `GetParameter`.

```
/**
* Initialize Systems 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 DynamoDB client with different credentials
    final AmazonDynamoDB client = AmazonDynamoDBClient.builder()
            .withCredentials(new AWSStaticCredentialsProvider(differentAWSCreds))
            .withRegion(getParameter("region")) //Getting configuration from
 Parameter Store
            .build();
    return client;
}

/**
* Helper method to retrieve parameter value
* @param parameterName identifier of the 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. Utilisez la commande `aws secretsmanager list-secrets` pour trouver les noms de vos secrets.

AWS CLI Exemple 1 : Référence à l'aide du nom du secret

Linux & macOS

```
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",
        "Type": "SecureString",
        "Value": "Fl*MEishm!al875",
        "Version": 0,
        "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\"
                }"
        "LastModifiedDate": 2018-05-14T21:47:14.743Z,
        "ARN": "arn:aws:secretsmanager:us-east-2:123456789012:secret:s1-secret-
               E18LRP",
      }
}
```

AWS CLI Exemple 2 : Référence qui inclut l'ID de la version

Linux & macOS

```
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",
        "Type": "SecureString",
        "Value": "Fl*MEishm!al875",
        "Version": 0,
        "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"
      }
        "LastModifiedDate": 2018-05-14T21:47:14.743Z,
        "ARN": "arn:aws:secretsmanager:us-east-2:123456789012:secret:s1-secret-
               E18LRP",
}
```

AWS CLI Exemple 3 : Référence qui inclut l'étape de la version

Linux & macOS

```
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",
        "Type": "SecureString",
        "Value": "Fl*MEishm!al875",
        "Version": 0,
        "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"
      }
```

```
      "LastModifiedDate": 2018-05-14T21:47:14.743Z,
      "ARN": "arn:aws:secretsmanager:us-east-2:123456789012:secret:s1-secret-
                E18LRP",
}
```

Utilisation de paramètres de Parameter Store dans les fonctions AWS Lambda

Parameter Store, une fonctionnalité de AWS Systems Manager, fournit un stockage hiérarchique sécurisé pour la gestion des données de configuration et la gestion des secrets. Vous pouvez stocker des données telles que des mots de passe, des chaînes de base de données, des ID d'Amazon Machine Image (AMI) et des codes de licence en tant que valeurs de paramètres.

Pour utiliser les paramètres Parameter Store des AWS Lambda fonctions sans utiliser de SDK, vous pouvez utiliser l'extension Lambda AWS Parameters and Secrets. Cette extension récupère les valeurs de paramètre et les met en cache pour une utilisation future. L'utilisation de l'extension Lambda peut réduire vos coûts en diminuant le nombre d'appels d'API vers Parameter Store. L'utilisation de l'extension peut également améliorer la latence, car la récupération d'un paramètre mis en cache est plus rapide que sa récupération depuis Parameter Store.

Une extension Lambda est un processus complémentaire qui améliore les capacités d'une fonction Lambda. Une extension est comme un client qui s'exécute en parallèle d'une invocation Lambda. Ce client parallèle peut s'interfacer avec votre fonction à tout moment au cours de son cycle de vie. Pour plus d'informations sur les extensions Lambda, veuillez consulter API Extensions Lambda dans le Guide du développeur AWS Lambda .

L'extension Lambda AWS Parameters and Secrets fonctionne à la fois pour etParameter Store. AWS Secrets Manager Pour savoir comment utiliser l'extension Lambda avec des secrets depuis Secrets Manager, consultez la section Utiliser des AWS Secrets Manager secrets dans les AWS Lambda fonctions du Guide de l'AWS Secrets Manager utilisateur.

Informations connexes

Utilisation de l'extension Lambda AWS Parameter and Secrets pour mettre en cache des paramètres et des secrets (AWS Compute Blog)

Fonctionnement de l'extension

Pour utiliser des paramètres dans une fonction Lambda sans extension Lambda, vous devez configurer votre fonction Lambda pour qu'elle reçoive des mises à jour de configuration en l'intégrant à l'action d'API `GetParameter` pour Parameter Store.

Lorsque vous utilisez l'extension Lambda AWS Parameters and Secrets, l'extension extrait la valeur du paramètre Parameter Store et la stocke dans le cache local. Ensuite, la valeur mise en cache est utilisée pour d'autres invocations jusqu'à son expiration. Les valeurs mises en cache expirent une fois leur valeur time-to-live (TTL) transmise. Vous pouvez configurer la valeur TTL à l'aide de la variable d'environnement `SSM_PARAMETER_STORE_TTL`, comme expliqué plus loin dans cette rubrique.

Si la TTL du cache configurée n'a pas expiré, la valeur du paramètre mise en cache est utilisée. Si le délai est expiré, la valeur mise en cache est invalidée et la valeur du paramètre est récupérée depuis Parameter Store.

En outre, le système détecte les valeurs de paramètres fréquemment utilisées et les conserve dans le cache tout en effaçant celles qui sont expirées ou inutilisées.

Détails de l'implémentation

Utilisez les informations suivantes pour vous aider à configurer l'extension Lambda AWS Parameters and Secrets.

Authentification

Pour autoriser et authentifier les requêtes de Parameter Store, l'extension utilise les mêmes informations d'identification que celles utilisées pour exécuter la fonction Lambda elle-même. Par conséquent, le rôle AWS Identity and Access Management (IAM) utilisé pour exécuter la fonction doit disposer des autorisations suivantes pour interagir avec Parameter Store :

- `ssm:GetParameter` : obligatoire pour récupérer les paramètres depuis Parameter Store
- `kms:Decrypt` : obligatoire si vous récupérez des paramètres `SecureString` depuis Parameter Store

Pour plus d'informations, veuillez consulter Rôle d'exécution AWS Lambda dans le Guide du développeur AWS Lambda .

Instanciation

Lambda instancie des instances distinctes correspondant au niveau de simultanéité requis par votre fonction. Chaque instance est isolée et conserve son propre cache local de vos données

de configuration. Pour plus d'informations sur les instances Lambda et la simultanéité, veuillez consulter Configuration de la simultanéité réservée Lambda dans le Guide du développeur AWS Lambda .

Aucune dépendance au kit SDK

L'extension Lambda AWS Parameters and Secrets fonctionne indépendamment de toute bibliothèque de langage du AWS SDK. Il n'est pas nécessaire de disposer d'un AWS SDK pour envoyer des requêtes GET àParameter Store.

Port du Localhost

Utilisez `localhost` dans vos requêtes GET. L'extension envoie des requêtes au port localhost 2773. Vous n'avez pas besoin de spécifier un point de terminaison externe ou interne pour utiliser l'extension. Vous pouvez configurer le port en définissant la `PARAMETERS_SECRETS_EXTENSION_HTTP_PORT`variable d'environnement.

Par exemple, dans Python, votre URL GET peut ressembler à l'exemple suivant.

```
parameter_url = ('http://localhost:' + port + '/systemsmanager/parameters/get/?
name=' + ssm_parameter_path)
```

Modifications apportées à la valeur d'un paramètre avant l'expiration de la TTL

L'extension ne détecte pas les modifications apportées à la valeur du paramètre et n'effectue pas d'actualisation automatique avant l'expiration de la TTL. Si vous modifiez la valeur d'un paramètre, les opérations utilisant la valeur de paramètre mise en cache peuvent échouer jusqu'à la prochaine actualisation du cache. Si vous prévoyez de modifier fréquemment la valeur d'un paramètre, nous vous recommandons de définir une valeur TTL plus courte.

Exigence d'en-tête

Pour récupérer des paramètres depuis le cache de l'extension, l'en-tête de votre requête GET doit inclure une référence `X-Aws-Parameters-Secrets-Token`. Définissez le jeton sur `AWS_SESSION_TOKEN`, qui est fourni par Lambda pour toutes les fonctions en cours d'exécution. L'utilisation de cet en-tête indique que l'appelant se trouve dans l'environnement Lambda.

Exemple

L'exemple suivant dans Python illustre une requête de base permettant de récupérer la valeur d'un paramètre mis en cache.

```
import urllib.request
```

```
import os
import json

aws_session_token = os.environ.get('AWS_SESSION_TOKEN')

def lambda_handler(event, context):
    # Retrieve /my/parameter from Parameter Store using extension cache
    req = urllib.request.Request('http://localhost:2773/systemsmanager/parameters/
get?name=%2Fmy%2Fparameter')
    req.add_header('X-Aws-Parameters-Secrets-Token', aws_session_token)
    config = urllib.request.urlopen(req).read()

    return json.loads(config)
```

Prise en charge par ARM

L'extension ne supporte pas du tout l'architecture ARM Régions AWS là où les x86 architectures x86_64 et sont prises en charge.

Pour la liste complète des ARN d'extension, veuillez consulter AWS Paramètres et secrets (ARN de l'extension Lambda).

Journalisation

Lambda enregistre les informations d'exécution relatives à l'extension ainsi qu'à la fonction à l'aide d'Amazon CloudWatch Logs. Par défaut, l'extension enregistre une quantité minimale d'informations dans CloudWatch. Pour journaliser plus d'informations, définissez la variable d'environnement `PARAMETERS_SECRETS_EXTENSION_LOG_LEVEL` sur `DEBUG`.

Ajout de l'extension à une fonction Lambda

Pour utiliser l'extension Lambda AWS Parameters and Secrets, vous devez l'ajouter à votre fonction Lambda sous forme de couche.

Utilisez l'une des méthodes suivantes pour ajouter l'extension à votre fonction.

AWS Management Console (Option d'ajout d'une couche)

1. Ouvrez la AWS Lambda console à l'adresse https://console.aws.amazon.com/lambda/.
2. Choisissez votre fonction. Dans la zone Layers (Couches), choisissez Add a layer (Ajouter une couche).
3. Dans la zone Choisir une couche, sélectionnez l'option Couches AWS .

4. Pour Couches AWS , sélectionnez AWS-Parameters-and-Secrets-Lambda-Extension, choisissez une version puis cliquez sur Ajouter.

AWS Management Console (Spécifiez l'option ARN)

1. Ouvrez la AWS Lambda console à l'adresse https://console.aws.amazon.com/lambda/.
2. Choisissez votre fonction. Dans la zone Layers (Couches), choisissez Add a layer (Ajouter une couche).
3. Dans la zone Choose a layer (Choisir une couche), sélectionnez l'option Specify an ARN (Spécifier un ARN).
4. Pour Spécifier un ARN, entrez l'ARN de l'extension pour votre architecture Région AWS et, puis choisissez Ajouter.

AWS Command Line Interface

Exécutez la commande suivante dans l' AWS CLI : Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

```
aws lambda update-function-configuration \
    --function-name function-name \
    --layers layer-ARN
```

Informations connexes

Utilisation de couches avec votre fonction Lambda

Configuration des extensions (archive de fichiers .zip)

AWS Paramètres et secrets Variables d'environnement de l'extension Lambda

Vous pouvez configurer l'extension en modifiant les variables d'environnement suivantes. Pour voir les paramètres actuels, définissez `PARAMETERS_SECRETS_EXTENSION_LOG_LEVEL` sur `DEBUG`. Pour plus d'informations, consultez la section Utilisation de variables d' AWS Lambda environnement dans le Guide du AWS Lambda développeur.

> **Note**
>
> AWS Lambda enregistre les détails des opérations relatives à l'extension Lambda et à la fonction Lambda dans Amazon Logs. CloudWatch

Variable d'environnement	Détails	Obligatoire	Valeurs valides	Valeur par défaut
`SSM_PARAMETER_STORE_TIMEOUT_MILLIS`	Délai d'expiration des requêtes adressées à Parameter Store, en millisecondes. La valeur 0 (zéro) indique qu'il n'y a pas de délai d'expiration.	Non	Tous les nombres entiers	0 (zéro)
`SECRETS_MANAGER_TIMEOUT_MILLIS`	Délai d'expiration des requêtes adressées à Secrets Manager, en millisecondes. La valeur 0 (zéro) indique qu'il n'y a pas de délai d'expiration.	Non	Tous les nombres entiers	0 (zéro)
`SSM_PARAMETER_STORE_TTL`	Durée de vie maximale valide, en secondes, d'un paramètre dans le cache avant qu'il ne soit invalidé. La valeur 0 (zéro)	Non	0 (zéro) à 300 s (cinq minutes)	300 s (cinq minutes)

Variable d'environnement	Détails	Obligatoire	Valeurs valides	Valeur par défaut
	indique que le cache doit être contourné . Cette variable est ignorée si la valeur de `PARAMETERS_SECRETS_EXTENSION_CACHE_SIZE` est 0 (zéro).			
`SECRETS_MANAGER_TTL`	Durée de vie maximale valide, en secondes, d'un secret dans le cache avant qu'il ne soit invalidé. La valeur 0 (zéro) indique que le cache est contourné. Cette variable est ignorée si la valeur de `PARAMETERS_SECRETS_EXTENSION_CACHE_SIZE` est 0 (zéro).	Non	0 (zéro) à 300 s (cinq minutes)	300 s (5 minutes)

Variable d'environnement	Détails	Obligatoire	Valeurs valides	Valeur par défaut
`PARAMETERS_SECRETS_EXTENSION_CACHE_ENABLED`	Détermine si le cache est activé pour l'extension. Valeur valeurs : `TRUE | FALSE`	Non	TRUE | FALSE	TRUE
`PARAMETERS_SECRETS_EXTENSION_CACHE_SIZE`	Taille maximale du cache en termes de nombre d'éléments. La valeur 0 (zéro) indique que le cache est contourné. Cette variable est ignorée si les deux valeurs TTL du cache sont égales à 0 (zéro).	Non	0 (zéro) à 1 000	1 000
`PARAMETERS_SECRETS_EXTENSION_HTTP_PORT`	Port du serveur HTTP local.	Non	1 - 65535	2773

Variable d'environnement	Détails	Obligatoire	Valeurs valides	Valeur par défaut
`PARAMETERS_SECRETS_EXTENSION_MAX_CONNECTIONS`	Nombre maximal de connexions pour les clients HTTP que l'extension utilise pour adresser des requêtes au Parameter Store ou à Secrets Manager. Il s'agit d'une configuration par client indiquant le nombre de connexions que le client Secrets Manager et le client Parameter Store établissent aux services backend.	Non	Minimum de 1 ; aucune limite maximale.	3

Variable d'environnement	Détails	Obligatoire	Valeurs valides	Valeur par défaut
`PARAMETERS_SECRETS_EXTENSION_LOG_LEVEL`	Le niveau de détail indiqué dans les journaux pour l'extension. Nous vous recommandons d'utiliser `DEBUG` pour obtenir le plus de détails sur la configuration de votre cache lorsque vous configurez et testez l'extension. Les journaux des opérations Lambda sont automatiquement transférés vers un groupe de journaux de CloudWatch journaux associé.	Non	`DEBUG | WARN | ERROR | NONE | INFO`	`INFO`

Exemples de commandes pour utiliser l' AWS Systems Manager Parameter Store et l'extension AWS Secrets Manager

Les exemples de cette section illustrent les actions d'API à utiliser avec l' AWS Secrets Manager extension AWS Systems Manager Parameter Store and.

Exemples de commandes pour Parameter Store

L'extension Lambda utilise un accès en lecture seule à l'action d'API. GetParameter

Pour appeler cette action, effectuez un appel HTTP GET similaire à ce qui suit.

```
GET http://localhost:port/systemsmanager/parameters/get?name=parameter-
path&version=version&label=label&withDecryption={true|false}
```

Dans cet exemple, *parameter-path* représente le nom complet du paramètre. *version* et *étiquette* sont les sélecteurs disponibles pour une utilisation avec l'`GetParameter`action. Ce format de commande permet d'accéder aux paramètres du niveau de paramètres standard.

> **Note**
>
> Lorsque vous utilisez des appels GET, les valeurs des paramètres doivent être codées pour que HTTP conserve les caractères spéciaux. Par exemple, au lieu de formater un chemin hiérarchique comme `/a/b/c`, codez des caractères qui pourraient être interprétés comme faisant partie de l'URL, tels que `%2Fa%2Fb%2Fc`.

```
GET http://localhost:port/systemsmanager/parameters/get/?name=MyParameter&version=5
```

Pour appeler un paramètre dans une hiérarchie, effectuez un appel HTTP GET similaire à ce qui suit.

```
GET http://localhost:port/systemsmanager/parameters/get?name=%2Fa%2Fb%2F&label=release
```

Pour appeler un paramètre public (global), effectuez un appel HTTP GET similaire à ce qui suit.

```
GET http://localhost:port/systemsmanager/parameters/get/?name=%2Faws%2Fservice%20list
%2F…
```

Pour effectuer un appel HTTP GET vers un secret Secrets Manager à l'aide de références Parameter Store, effectuez un appel HTTP GET similaire à ce qui suit.

```
GET http://localhost:port/systemsmanager/parameters/get?name=%2Faws%2Freference
%2Fsecretsmanager%2F…
```

Pour effectuer un appel en utilisant l'Amazon Resource Name (ARN) pour un paramètre, effectuez un appel HTTP GET similaire à ce qui suit.

```
GET http://localhost:port/systemsmanager/parameters/get?name=arn:aws:ssm:us-
east-1:123456789012:parameter/MyParameter
```

Pour effectuer un appel qui accède à un paramètre `SecureString` avec le déchiffrement, effectuez un appel HTTP GET similaire à ce qui suit.

```
GET http://localhost:port/systemsmanager/parameters/get?
name=MyParameter&withDecryption=true
```

Vous pouvez spécifier que les paramètres ne sont pas déchiffrés en omettant `withDecryption` ou en le définissant explicitement sur `false`. Vous pouvez également spécifier une version ou une étiquette, mais pas les deux. Le cas échéant, seule la première d'entre elles placée après le point d'interrogation (?) dans l'URL est utilisée.

AWS Paramètres et secrets (ARN de l'extension Lambda)

Les tableaux suivants fournissent des ARN d'extension pour les architectures et les régions prises en charge.

Rubriques

- ARN d'extension pour les architectures x86_64 et x86
- ARN d'extension pour les architectures ARM64 et les architectures Mac with Apple silicon

ARN d'extension pour les architectures x86_64 et x86

Région	ARN
USA Est (Ohio)	`arn:aws:lambda:us-east-2:590474943231:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
USA Est (Virginie du Nord)	`arn:aws:lambda:us-east-1:177933569100:layer:AWS-Parame`

Région	ARN
	ters-and-Secrets-Lambda-Ext ension:11
USA Ouest (Californie du Nord)	arn:aws:lambda:us-west-1:99 7803712105:layer:AWS-Parame ters-and-Secrets-Lambda-Ext ension:11
USA Ouest (Oregon)	arn:aws:lambda:us-west-2:34 5057560386:layer:AWS-Parame ters-and-Secrets-Lambda-Ext ension:11
Afrique (Le Cap)	arn:aws:lambda:af-south-1:3 17013901791:layer:AWS-Param eters-and-Secrets-Lambda-Ex tension:11
Asie-Pacifique (Hong Kong)	arn:aws:lambda:ap-east-1:76 8336418462:layer:AWS-Parame ters-and-Secrets-Lambda-Ext ension:11
Région Asie-Pacifique (Hyderabad)	arn:aws:lambda:ap-south-2:0 70087711984:layer:AWS-Param eters-and-Secrets-Lambda-Ex tension:8
Asie-Pacifique (Jakarta)	arn:aws:lambda:ap-southeast -3:490737872127:layer:AWS-P arameters-and-Secrets-Lambda- Extension:11

Région	ARN
Asie-Pacifique (Melbourne)	`arn:aws:lambda:ap-southeast-4:090732460067:layer:AWS-Parameters-and-Secrets-Lambda-Extension:1`
Asie-Pacifique (Mumbai)	`arn:aws:lambda:ap-south-1:176022468876:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Asie-Pacifique (Osaka)	`arn:aws:lambda:ap-northeast-3:576959938190:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Asia Pacific (Seoul)	`arn:aws:lambda:ap-northeast-2:738900069198:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Asie-Pacifique (Singapour)	`arn:aws:lambda:ap-southeast-1:044395824272:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Asie-Pacifique (Sydney)	`arn:aws:lambda:ap-southeast-2:665172237481:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Asie-Pacifique (Tokyo)	`arn:aws:lambda:ap-northeast-1:133490724326:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`

Région	ARN
Canada (Centre)	`arn:aws:lambda:ca-central-1:200266452380:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Canada Ouest (Calgary)	`arn:aws:lambda:ca-west-1:243964427225:layer:AWS-Parameters-and-Secrets-Lambda-Extension:1`
Chine (Beijing)	`arn:aws-cn:lambda:cn-north-1:287114880934:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
China (Ningxia)	`arn:aws-cn:lambda:cn-northwest-1:287310001119:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Europe (Francfort)	`arn:aws:lambda:eu-central-1:187925254637:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Europe (Irlande)	`arn:aws:lambda:eu-west-1:015030872274:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Europe (Londres)	`arn:aws:lambda:eu-west-2:133256977650:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`

Région	ARN
Europe (Milan)	`arn:aws:lambda:eu-south-1:325218067255:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Europe (Paris)	`arn:aws:lambda:eu-west-3:780235371811:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Région Europe (Espagne)	`arn:aws:lambda:eu-south-2:524103009944:layer:AWS-Parameters-and-Secrets-Lambda-Extension:8`
Europe (Stockholm)	`arn:aws:lambda:eu-north-1:427196147048:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Israël (Tel Aviv)	`arn:aws:lambda:il-central-1:148806536434:layer:AWS-Parameters-and-Secrets-Lambda-Extension:1`
Région Europe (Zurich)	`arn:aws:lambda:eu-central-2:772501565639:layer:AWS-Parameters-and-Secrets-Lambda-Extension:8`
Moyen-Orient (Bahreïn)	`arn:aws:lambda:me-south-1:832021897121:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`

Région	ARN
Moyen-Orient (EAU)	`arn:aws:lambda:me-central-1:858974508948:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
Amérique du Sud (São Paulo)	`arn:aws:lambda:sa-east-1:933737806257:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
AWS GovCloud (USA Est)	`arn:aws-us-gov:lambda:us-gov-east-1:129776340158:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`
AWS GovCloud (US-Ouest)	`arn:aws-us-gov:lambda:us-gov-west-1:127562683043:layer:AWS-Parameters-and-Secrets-Lambda-Extension:11`

ARN d'extension pour les architectures ARM64 et les architectures Mac with Apple silicon

Région	ARN
USA Est (Ohio)	`arn:aws:lambda:us-east-2:590047494323 1:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:11`
USA Est (Virginie du Nord)	`arn:aws:lambda:us-east-1:177933569100:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:11`

Région	ARN
Région US West (N. California)	`arn:aws:lambda:us-west-1:997803712105:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:8`
USA Ouest (Oregon)	`arn:aws:lambda:us-west-2:345057560386:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:11`
Région Afrique (Le Cap)	`arn:aws:lambda:af-south-1:317013901791:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:8`
Région Asie-Pacifique (Hong Kong)	`arn:aws:lambda:ap-east-1:768336418462:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:8`
Région Asie-Pacifique (Jakarta)	`arn:aws:lambda:ap-southeast-3:490737872127:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:8`
Asie-Pacifique (Mumbai)	`arn:aws:lambda:ap-south-1:176022468876:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:11`
Asie-Pacifique (Osaka)	`arn:aws:lambda:ap-northeast-3:576959938190:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:8`

Région	ARN
Région Asia Pacific (Seoul)	`arn:aws:lambda:ap-northeast-2:738900069198:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:8`
Asie-Pacifique (Singapour)	`arn:aws:lambda:ap-southeast-1:044395824272:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:11`
Asie-Pacifique (Sydney)	`arn:aws:lambda:ap-southeast-2:665172237481:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:11`
Asia Pacific (Tokyo)	`arn:aws:lambda:ap-northeast-1:133490724326:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:11`
Région Canada (Centre)	`arn:aws:lambda:ca-central-1:200266452380:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:8`
Europe (Francfort)	`arn:aws:lambda:eu-central-1:187925254637:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:11`
Europe (Irlande)	`arn:aws:lambda:eu-west-1:015030872274:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:11`

Région	ARN
Europe (Londres)	`arn:aws:lambda:eu-west-2:133256977650:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:11`
Europe (Milan) Region	`arn:aws:lambda:eu-south-1:325218067255:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:8`
Région Europe (Paris)	`arn:aws:lambda:eu-west-3:780235371811:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:8`
Région Europe (Stockholm)	`arn:aws:lambda:eu-north-1:427196147048:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:8`
Middle East (Bahrain) Region	`arn:aws:lambda:me-south-1:832021897121:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:8`
Région Amérique du Sud (São Paulo)	`arn:aws:lambda:sa-east-1:933737806257:layer:AWS-Parameters-and-Secrets-Lambda-Extension-Arm64:8`

Intégration à d'autres produits et services

AWS Systems Manager dispose d'une intégration intégrée pour les produits et services présentés dans le tableau suivant.

Ansible	Ansibleest une plateforme d'automatisation informatique qui facilite le déploiement de vos applications et de vos systèmes. Systems Manager fournit le document Systems Manager (document SSM) `AWS-ApplyAnsiblePlaybooks` qui vous permet de créer des State Manager associations qui exécutent des Ansible playbooks. En savoir plus Procédure pas à pas : création d'associations qui exécutent Ansible des playbooks
Chef	Chefest un outil d'automatisation informatique qui facilite le déploiement de vos applications et de vos systèmes. Systems Manager fournit le document `AWS-ApplyChefRecipes` SSM, qui vous permet de créer des associationsState Manager, une fonctionnalité de AWS Systems Manager, qui exécutent des Chef recettes. En savoir plus Procédure pas à pas : création d'associations qui exécutent Chef des recettes Systems Manager s'intègre également aux Chef InSpecprofils, ce qui vous permet d'exécuter des analyses de conformité et de visualiser les nœuds conformes et non conformes. En savoir plus

	Utilisation de Chef InSpec profils avec Systems Manager Compliance
GitHub	GitHubfournit un hébergement pour le développement de logiciels, le contrôle des versions et la collaboration. Systems Manager fournit le document SSMAWS-RunDocument , qui vous permet d'exécuter d'autres documents SSM stockés dansGitHub, et le document SSMAWS-RunRemoteScript , qui vous permet d'exécuter des scripts stockés dans. GitHub En savoir plus • Exécution de documents à partir d'emplacements distants • Exécution de scripts depuis GitHub
Jenkins	Jenkinsest un serveur d'automatisation open source qui permet aux développeurs de créer, de tester et de déployer leurs logiciels de manière fiable. Automation, une fonctionnalité de Systems Manager, peut être utilisée comme une étape post-création pour préinstaller des versions d'applications dans Amazon Machine Images (AMIs). En savoir plus Mise à jour AMIs grâce à l'automatisation et Jenkins

ServiceNow	ServiceNowest un système de gestion des services d'entreprise qui vous permet de gérer vos services et opérations informatiques. AutomationChange Manager, Incident Manager et OpsCenter toutes les fonctionnalités de Systems Manager s'intègrent à ServiceNo w l'aide du connecteur de gestion des AWS services. Grâce à cette intégration, vous pouvez consulter, créer, mettre à jour, ajouter de la correspondance et résoudre des AWS Support cas à partir deServiceNow. En savoir plus Intégration avec ServiceNow

Rubriques

- Exécution de scripts depuis GitHub
- Utilisation de Chef InSpec profils avec Systems Manager Compliance
- Intégration avec ServiceNow

Exécution de scripts depuis GitHub

Cette rubrique explique comment utiliser le document prédéfini Systems Manager (document SSM) `AWS-RunRemoteScript` pour télécharger des scriptsGitHub, notamment depuis Ansible Playbooks, Python, Ruby et des scripts. PowerShell En utilisant ce document SSM, vous n'avez plus besoin de porter manuellement des scripts vers Amazon Elastic Compute Cloud (Amazon EC2) ou de les encapsuler dans des documents SSM. AWS Systems Manager l'intégration avec GitHub promeut l'infrastructure sous forme de code, ce qui réduit le temps nécessaire à la gestion des nœuds tout en normalisant les configurations au sein de votre flotte.

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 Création de documents composites.

Vous pouvez également télécharger un répertoire contenant plusieurs scripts. Lorsque vous exécutez le script principal du répertoire, Systems Manager exécute également 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 Manager ne vérifie pas que le script est à même de s'exécuter sur un nœud. Avant de télécharger et d'exécuter le script, vérifiez que le logiciel requis est installé sur le nœud. Vous pouvez également créer un document composite qui installe le logiciel par l'intermédiaire de Run Command ou de State Manager, des fonctionnalités de AWS Systems Manager, puis télécharge et exécute le script.
- Il est de votre responsabilité 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.
- GitHub Enterpriseles référentiels ne sont pas pris en charge.

Rubriques

Exécutez des Ansible Playbooks à partir de GitHub

Cette section inclut des procédures qui vous aideront à exécuter Ansible des Playbooks à GitHub l'aide de la console ou du AWS Command Line Interface (AWS CLI).

Avant de commencer

Si vous envisagez d'exécuter un script stocké dans un GitHub dépôt privé, créez un AWS Systems Manager `SecureString` paramètre pour votre jeton d'accès GitHub sécurisé. Vous ne pouvez pas accéder à un script dans un GitHub dépôt privé en passant 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.

Exécuter un Ansible Playbook depuis GitHub (console)

Exécutez un Ansible Playbook depuis GitHub

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Run Command.
3. Sélectionnez Run Command (Exécuter la commande).
4. In the Command document (Document de commande), sélectionnez **`AWS-RunRemoteScript`**.
5. Dans Paramètres de la commande, procédez comme suit :

   - Dans Type de source, sélectionnez GitHub.
   - Dans la zone Source Info (Infos sur la source), saisissez les informations requises pour accéder à la source en respectant le format suivant.

   ```
   {
     "owner": "owner_name",
     "repository": "repository_name",
     "getOptions": "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",
       "getOptions": "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 un commit particulier de votre référentiel, utilisez `commitID` avec `getOptions` au lieu de `branch`. Par exemple :
`"getOptions": "commitID:bbc1ddb94...b76d3bEXAMPLE",`

- Dans le champ Command Line (Ligne de commande), saisissez les paramètres d'exécution du script. Voici un exemple.

  `ansible-playbook -i “localhost,” --check -c local webserver.yml`

- (Facultatif) Dans le champ Working Directory (Répertoire de travail), saisissez le nom d'un répertoire du nœud 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 Targets (Cibles), sélectionnez les nœuds gérés sur lesquels vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant des instances ou des appareils de périphérie manuellement ou en spécifiant un groupe de ressources.

   Tip

   Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés 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. Pour Rate control (Contrôle de débit) :

   - Dans Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage de nœuds gérés sur lesquels exécuter simultanément la commande.

> **Note**
>
> Si vous avez sélectionné des cibles en spécifiant les balises appliquées aux nœuds gérés ou en spécifiant des groupes de ressources AWS , et que vous n'êtes pas certain du nombre de nœuds gérés ciblés, limitez le nombre de cibles autorisées à exécuter simultanément le document en indiquant un pourcentage.

- Dans Error threshold (Seuil d'erreur), indiquez quand arrêter l'exécution de la commande sur les autres nœuds gérés après l'échec de celle-ci sur un certain nombre ou un certain pourcentage de nœuds. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les nœuds gérés sur lesquels la commande est toujours 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). Saisissez 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 (pour les instances EC2) ou de la fonction du service IAM (pour les machines activées par un système hybride) attribués à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, assurez-vous que le profil d'instance ou le rôle de service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

10. Dans la section SNS notifications (Notifications SNS), si vous souhaitez envoyer des notifications sur le statut d'exécution des commandes, cochez la case Enable SNS notifications (Activer les notifications SNS).

    Pour plus d'informations sur la configuration des notifications Amazon SNS pour Run Command, consultez Surveillance des changements d'état du Systems Manager à l'aide des notifications Amazon SNS.

11. Cliquez sur Exécuter.

Exécutez un Ansible Playbook à partir GitHub de AWS CLI

1. Installez et configurez le AWS Command Line Interface (AWS CLI), si ce n'est pas déjà fait.

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l' AWS CLI.
2. Exécutez la commande suivante pour télécharger et exécuter un script depuisGitHub.

```
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"]}'
```

Exécutez des scripts Python à partir de GitHub

Cette section inclut des procédures pour vous aider à exécuter des scripts Python GitHub à l'aide de la AWS Systems Manager console ou du AWS Command Line Interface (AWS CLI).

Exécuter un script Python depuis GitHub (console)

Exécutez un script Python depuis GitHub

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.

2. Dans le panneau de navigation, sélectionnez Run Command.
3. Sélectionnez Run Command (Exécuter la commande).
4. In the Command document (Document de commande), sélectionnez **`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), saisissez les informations requises pour accéder à la source en respectant le format suivant.

   ```
   {
       "owner": "owner_name",
       "repository": "repository_name",
       "getOptions": "branch:branch_name",
       "path": "path_to_document",
       "tokenInfo": "{{ssm-secure:SecureString_parameter_name}}"
   }
   ```

   Par exemple, le répertoire suivant télécharge un répertoire de scripts nommé complex-script.

   ```
   {
       "owner": "TestUser1",
       "repository": "SSMTestDocsRepo",
       "getOptions": "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 un commit particulier de votre référentiel, utilisez `commitID` avec `getOptions` au lieu de `branch`. Par exemple :
   > `"getOptions": "commitID:bbc1ddb94...b76d3bEXAMPLE",`

   - Dans le champ Command Line (Ligne de commande), saisissez les paramètres pour l'exécution du script. Voici un exemple.

```
mainFile.py argument-1 argument-2
```

Cet exemple exécute `mainFile.py`, qui peut ensuite exécuter d'autres scripts du répertoire `complex-script`.

- (Facultatif) Dans le champ Working Directory (Répertoire de travail), saisissez le nom d'un répertoire du nœud 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 Targets (Cibles), sélectionnez les nœuds gérés sur lesquels vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant des instances ou des appareils de périphérie manuellement ou en spécifiant un groupe de ressources.

> **Tip**
>
> Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés 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. Pour Rate control (Contrôle de débit) :

   - Dans Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage de nœuds gérés sur lesquels exécuter simultanément la commande.

> **Note**
>
> Si vous avez sélectionné des cibles en spécifiant les balises appliquées aux nœuds gérés ou en spécifiant des groupes de ressources AWS , et que vous n'êtes pas certain du nombre de nœuds gérés ciblés, limitez le nombre de cibles autorisées à exécuter simultanément le document en indiquant un pourcentage.

- Dans Error threshold (Seuil d'erreur), indiquez quand arrêter l'exécution de la commande sur les autres nœuds gérés après l'échec de celle-ci sur un certain nombre ou un certain pourcentage de nœuds. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse d'envoyer la commande à la réception de la quatrième erreur. Les nœuds gérés sur lesquels la commande est toujours 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). Saisissez 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 (pour les instances EC2) ou de la fonction du service IAM (pour les machines activées par un système hybride) attribués à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, assurez-vous que le profil d'instance ou le rôle de service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

10. Dans la section SNS notifications (Notifications SNS), si vous souhaitez envoyer des notifications sur le statut d'exécution des commandes, cochez la case Enable SNS notifications (Activer les notifications SNS).

    Pour plus d'informations sur la configuration des notifications Amazon SNS pour Run Command, consultez Surveillance des changements d'état du Systems Manager à l'aide des notifications Amazon SNS.

11. Cliquez sur Exécuter.

Exécutez un script Python à GitHub l'aide du AWS CLI

1. Installez et configurez le AWS Command Line Interface (AWS CLI), si ce n'est pas déjà fait.

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l' AWS CLI.

2. Exécutez la commande suivante pour télécharger et exécuter un script depuisGitHub.

```
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.

```
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 nommé `complex-script`. L'entrée `commandLine` exécute `mainFile.py`, qui peut ensuite exécuter d'autres scripts du répertoire `complex-script`.

Utilisation de Chef InSpec profils avec Systems Manager Compliance

AWS Systems Manager s'intègre à Chef InSpec. Chef InSpecest un framework de test open source qui vous permet de créer des profils lisibles par l'homme dans lesquels stocker ou dans GitHub Amazon Simple Storage Service (Amazon S3). Ensuite, vous pouvez utiliser Systems Manager pour exécuter des analyses de conformité et afficher les nœuds conformes et non conformes. Un profil est une exigence en termes de sécurité, de conformité ou de politique pour votre environnement informatique. Par exemple, vous pouvez créer des profils qui effectuent les vérifications suivantes lorsque vous analysez vos nœuds à l'aide de la fonctionnalité Compliance d' AWS 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 des InSpec profils pour les instances Amazon Elastic Compute Cloud (Amazon EC2) et les serveurs sur site ou les machines virtuelles (VM) que vous gérez avec Systems Manager. L'exemple de Chef InSpec profil 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
```

InSpec inclut un ensemble de ressources qui vous aident à rédiger rapidement des chèques et des contrôles d'audit. InSpec utilise le langage DSL (InSpec Domain-Specific Language) pour écrire ces contrôles en Ruby. Vous pouvez également utiliser des profils créés par une large communauté d' InSpec utilisateurs. Par exemple, le projet DevSec Chef-os-Hardening GitHub inclut des dizaines de profils pour vous aider à sécuriser vos nœuds. Vous pouvez créer et stocker des profils dans GitHub Amazon S3.

Comment ça marche

Voici comment fonctionne le processus d'utilisation des InSpec profils avec Compliance :

1. Identifiez les InSpec profils prédéfinis que vous souhaitez utiliser ou créez les vôtres. Vous pouvez utiliser des profils prédéfinis GitHub pour commencer. Pour plus d'informations sur la création de vos propres InSpec profils, consultez la section Chef InSpecProfils Chef.
2. Stockez les profils dans un GitHub référentiel public ou privé, ou dans un compartiment S3.
3. Exécutez Compliance avec vos InSpec profils à l'aide du document Systems Manager (document SSM). `AWS-RunInspecChecks` Vous pouvez démarrer une analyse de conformité en utilisant Run Command une fonctionnalité de AWS Systems Manager, pour les analyses à la demande, ou vous pouvez planifier des analyses de conformité régulières en utilisant State Manager une fonctionnalité de AWS Systems Manager.
4. Identifiez les nœuds non conformes à l'aide de l'API Compliance ou de la console de la fonctionnalité Compliance.

> **Note**
>
> Notez les informations suivantes.
>
> - Chefutilise un client sur vos nœuds 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. Dans le cas contraire, Systems Manager installe le Chef client pendant le scan, puis le désinstalle une fois le scan terminé.
> - L'exécution du document SSM `AWS-RunInspecChecks`, comme décrit dans cette rubrique, affecte une entrée de conformité de type `Custom:Inspec` à chaque nœud ciblé. Pour attribuer ce type de conformité, le document appelle l'opération PutComplianceItems API.

Exécution d'une analyse InSpec de conformité

Cette section contient des informations sur la manière d'exécuter une analyse de InSpec conformité à l'aide de la console Systems Manager et du AWS Command Line Interface (AWS CLI). La procédure pour la console vous montre comment configurer State Manager pour exécuter l'analyse. La AWS CLI procédure indique comment configurer Run Command pour exécuter le scan.

Exécution d'une analyse de InSpec conformité avec State Manager (console)

Pour exécuter une analyse InSpec de conformité à State Manager l'aide de la AWS Systems Manager console

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez State Manager.
3. Sélectionnez Créer une association.
4. Dans la section Fournir les détails de l'association, entrez un nom.
5. Dans la liste Document, sélectionnez **`AWS-RunInspecChecks`**.
6. Dans la liste Version du document, sélectionnez Dernière version lors de l'exécution.
7. Dans la section Paramètres, dans la liste Type de source, choisissez S3 GitHubou S3.

   Si vous le souhaitez GitHub, entrez le chemin d'accès à un InSpec profil dans un GitHub référentiel public ou privé dans le champ Source Info. Voici un exemple de chemin vers un

profil public fourni par l'équipe de 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, entrez une URL valide vers un InSpec profil d'un compartiment S3 dans le champ Source Info.

Pour plus d'informations sur la manière dont Systems Manager s'intègre GitHub à Amazon S3, consultezExécution de scripts depuis GitHub.

8. Dans la section Targets (Cibles), sélectionnez les nœuds gérés sur lesquels vous souhaitez exécuter cette opération en spécifiant des balises, en sélectionnant des instances ou des appareils de périphérie manuellement ou en spécifiant un groupe de ressources.

> **Tip**
>
> Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés pour obtenir des conseils de dépannage.

9. Dans la section Spécifier le programme, utilisez les options de générateur de planification pour créer un calendrier d'exécution des analyses de conformité.
10. Pour Rate control (Contrôle de débit) :

    - Dans Concurrency (Simultanéité), spécifiez un nombre ou un pourcentage de nœuds gérés sur lesquels exécuter simultanément la commande.

    > **Note**
    >
    > Si vous avez sélectionné des cibles en spécifiant les balises appliquées aux nœuds gérés ou en spécifiant des groupes de ressources AWS , et que vous n'êtes pas certain du nombre de nœuds gérés ciblés, limitez le nombre de cibles autorisées à exécuter simultanément le document en indiquant un pourcentage.

    - Dans Error threshold (Seuil d'erreur), indiquez quand arrêter l'exécution de la commande sur les autres nœuds gérés après l'échec de celle-ci sur un certain nombre ou un certain pourcentage de nœuds. Si, par exemple, vous spécifiez trois erreurs, Systems Manager cesse

d'envoyer la commande à la réception de la quatrième erreur. Les nœuds gérés sur lesquels la commande est toujours 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). Saisissez 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 (pour les instances EC2) ou de la fonction du service IAM (pour les machines activées par un système hybride) attribués à l'instance, et non celles de l'utilisateur IAM qui effectue cette tâche. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager ou Créer un rôle de service IAM pour un environnement hybride. En outre, si le compartiment S3 spécifié se trouve dans un autre compartiment Compte AWS, assurez-vous que le profil d'instance ou le rôle de service IAM associé au nœud géré dispose des autorisations nécessaires pour écrire dans ce compartiment.

12. Sélectionnez 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 sélectionnez Conformité dans le panneau de navigation.
14. Dans Corresponding managed instances (Instances gérées correspondantes), recherchez les nœuds pour lesquels la colonne Compliance Type (Type de conformité) contient la valeur Custom:Inspec.
15. Sélectionnez un ID de nœud pour afficher le détail des états non conformes.

Exécution d'une analyse de InSpec conformité avec Run Command (AWS CLI)

1. Installez et configurez le AWS Command Line Interface (AWS CLI), si ce n'est pas déjà fait.

   Pour de plus amples informations, consultez Installation ou mise à jour de la version la plus récente de l' AWS CLI.
2. Exécutez l'une des commandes suivantes pour exécuter un InSpec profil depuis Amazon S3 GitHub ou depuis Amazon S3.

La commande d' utilise les paramètres suivants :

- SourceType : ou Amazon S3 GitHub
- SourceInfo : URL du dossier de InSpec profil situé dans un compartiment S3 GitHub ou dans 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.

```
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.aws-api-domain/DOC-EXAMPLE-
BUCKET/Inspec.yml_file\"}"]}'
```

Voici un exemple.

```
aws ssm send-command --document-name "AWS-RunInspecChecks" --targets
 '[{"Key":"tag:testEnvironment","Values":["webServers"]}]' --
parameters'{"sourceType":["S3"],"sourceInfo":["{\"path\":\"https://s3.aws-api-
domain/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 le détail d'un nœud non conforme.

```
aws ssm list-compliance-items --resource-ids node_ID --resource-type
 ManagedInstance --filters Key=DocumentName,Values=AWS-RunInspecChecks
```

Intégration avec ServiceNow

ServiceNowfournit un système de gestion des services basé sur le cloud pour créer et gérer des flux de travail au niveau de l'organisation, tels que pour les services informatiques, les systèmes de billetterie et le support. Le connecteur de gestion des AWS services s'ServiceNowintègre à Systems Manager pour approvisionner, gérer et exploiter les AWS ressources à partir deServiceNow. Vous pouvez utiliser le connecteur de gestion des AWS services pour intégrer ServiceNow AutomationChange Manager, Incident Manager et OpsCenter toutes les fonctionnalités de AWS Systems Manager.

Vous pouvez effectuer les tâches suivantes à l'aide de ServiceNow :

- Exécutez des playbooks d'automatisation à partir de Systems Manager.
- Consultez, mettez à jour et résolvez les incidents à partir de Systems Manager OpsItems.
- Consultez et gérez les éléments opérationnels, tels que les incidents, via Systems Manager OpsCenter.
- Consultez et exécutez les demandes de modification Systems Manager à partir d'une liste organisée de modèles de modification pré-approuvés.
- Gérez et résolvez les incidents impliquant des applications AWS hébergées en intégrant Incident Manager.

> **Note**
>
> Pour plus d'informations sur la manière d'intégrerServiceNow, consultez la section Configuration des intégrations AWS de services dans le Guide de l'administrateur du connecteur de gestion des AWS services.

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 nœuds gérés de votre compte qui vous aident à suivre ou à cibler les nœuds par système d'exploitation et environnement, telles que vos nœuds SUSE Linux Enterprise Server regroupés sous `development`, `staging` et `production`. Vous pouvez également effectuer des opérations sur des ressources en spécifiant cette paire clé-valeur dans vos commandes, par exemple en exécutant un script de mise à jour sur tous les nœuds du groupe ou en vérifiant l'état de ces nœuds.

Vous pouvez utiliser les balises appliquées à vos ressources AWS Systems Manager dans diverses opérations. Par exemple, vous pouvez cibler uniquement les nœuds gérés qui sont balisés avec une paire clé-valeur de balise spécifiée lorsque vous exécutez une commande ou affectez des cibles à une fenêtre de maintenance. Vous pouvez également restreindre l'accès à vos ressources 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 Resource Groups (Groupes de ressources) 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 plus d'informations, consultez Qu'est-ce que l'AWS Resource Groups ? dans le Guide de l'utilisateur AWS Resource Groups.

Le reste de ce chapitre décrit comment ajouter et supprimer des balises des ressources Systems Manager.

Rubriques

- Balisage des associations Systems Manager
- Automatisations de balisage
- Balisage des documents Systems Manager
- Balisage des fenêtres de maintenance
- Balisage des nœuds gérés
- Balisage d'OpsItems
- Balisage de paramètres Systems Manager
- Balisage des références de correctifs

Les ressources Systems Manager susceptibles d'être balisées

Vous pouvez appliquer des balises aux ressources suivantes dans AWS Systems Manager :

- Associations
- Automatisations
- Documents
- Fenêtres de maintenance
- Nœuds gérés
- OpsItems
- OpsMetadata
- Paramètres
- Références de correctifs

Vous pouvez ajouter chacun de ces types, à l'exception d'OpsItems et d'OpsMetadata, 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 de nœuds gérés, puis exécuter une tâche de fenêtre de maintenance qui cible uniquement les nœuds avec cette paire clé-valeur.

Vous pouvez également restreindre l'accès des utilisateurs à ces types de ressources en créant des politiques AWS Identity and Access Management (IAM) qui spécifient les balises auxquelles un utilisateur peut accéder et en attachant la politique à des entités IAM (utilisateurs, rôles ou groupes). 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 (documents SSM), puis créer et appliquer une politique IAM 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 politiques IAM qui limitent les utilisateurs ou groupes qui sont autorisés à afficher ou à mettre à jour ces ressources. Par exemple, le directeur d'une organisation pourrait bénéficier d'un accès complet à tous les types de ressources OpsItems, mais les développeurs de logiciels et les ingénieurs support pourraient uniquement avoir 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 politique IAM qui accorde l'accès uniquement à ces ressources, par exemple `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 le travail des utilisateurs aux ressources de ProjectA dans l'environnement de développement.

Vous pouvez utiliser la console de Groupes de ressources Systems Manager, la console pour les types de ressources pris en charge (par exemple, la console Maintenance Windows ou la console OpsCenter), la AWS Command Line Interface (AWS CLI) et les AWS Tools for PowerShell. Vous pouvez ajouter des balises lorsque vous créez ou mettez à jour une ressource. Par exemple, vous pouvez utiliser la commande AWS CLI add-tags-to-resource 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 des associations Systems Manager

Les rubriques de cette section décrivent la méthode d'utilisation des balises relatives aux State Manager associations. State Manager est une composante de AWS Systems Manager.

Rubriques

Création des associations avec des balises.

Il est possible d'ajouter des balises à un State Manager association lors de sa création à l'aide deAWS CLI. L'ajout de balises à une association à l'aide de la console du System Manager n'est point pris en charge. Pour plus d'informations, consultez Créer une association (ligne de commande).

L'ajout de balises à une association existante

Utilisez les procédures suivantes afin d'ajouter des balises à un State Manager association à l'aide de la ligne de commande.

Rubriques

- L'ajout de balises à une association existante (AWS CLI)
- L'ajout de balises à une association existante (AWS Tools for PowerShell)

L'ajout de balises à une association existante (AWS CLI)

1. À l'aide de AWS CLI, exécutez la commande suivante afin de répertorier les associations balisables.

   ```
   aws ssm list-associations
   ```

   Notez le nom de l'association que vous souhaitez baliser.

2. Exécutez la commande suivante afin de baliser une association. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   ```
   aws ssm add-tags-to-resource \
       --resource-type "Association" \
       --resource-id "association-ID" \
       --tags "Key=tag-key,Value=tag-value"
   ```

   En cas de réussite, la commande n'a aucune sortie.

3. Exécutez la commande suivante pour la vérification des balises de l'association.

   ```
   aws ssm list-tags-for-resource --resource-type "Association" --resource-id
    "association-ID"
   ```

L'ajout de balises à une association existante (AWS Tools for PowerShell)

1. Exécutez la commande suivante pour l'établissement d'une liste des paramètres auxquels vous pouvez attribuer une balise.

```
Get-SSMAssociationList
```

2. Exécutez les commandes suivantes pour attribuer une balise à un paramètre. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

```
$tag = New-Object Amazon.SimpleSystemsManagement.Model.Tag
```

```
$tag.Key = "tag-key"
```

```
$tag.Value = "tag-value"
```

```
Add-SSMResourceTag `
    -ResourceType "Association" `
    -ResourceId "association-ID" `
    -Tag $tag `
    -Force
```

3. Exécutez la commande suivante pour la vérification des balises de l'association.

```
Get-SSMResourceTag `
    -ResourceType "Association" `
    -ResourceId "association-ID"
```

Suppression de balises dans une association

Vous pouvez utiliser la ligne de commande afin de supprimer des balises dans un State Manager association.

Suppression de balises dans une association (ligne de commande)

1. À l'aide de votre outil de ligne de commande préféré, exécutez la commande suivante afin de répertorier les associations de votre compte.

Linux & macOS

```
aws ssm list-associations
```

Windows

```
aws ssm list-associations
```

PowerShell

```
Get-SSMAssociationList
```

Notez le nom de l'association dont vous souhaitez la suppression des balises.

2. Exécutez la commande suivante pour la suppression des balises d'une association . Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

Linux & macOS

```
aws ssm remove-tags-from-resource \
    --resource-type "Association" \
    --resource-id "association-ID" \
    --tag-key "tag-key"
```

Windows

```
aws ssm remove-tags-from-resource ^
    --resource-type "Association" ^
    --resource-id "association-ID" ^
    --tag-key "tag-key"
```

PowerShell

```
Remove-SSMResourceTag
    -ResourceId "association-ID"
    -ResourceType "Association"
    -TagKey "tag-key
```

En cas de réussite, la commande n'a aucune sortie.

3. Exécutez la commande suivante pour la vérification des balises de l'association.

   Linux & macOS

   ```
   aws ssm list-tags-for-resource \
       --resource-type "Association" \
       --resource-id "association-ID"
   ```

   Windows

   ```
   aws ssm list-tags-for-resource ^
       --resource-type "Association" ^
       --resource-id "association-ID"
   ```

   PowerShell

   ```
   Get-SSMResourceTag `
       -ResourceType "Association" `
       -ResourceId "association-ID"
   ```

Automatisations de balisage

Les rubriques dans cette section décrivent comment utiliser les balises en automatisation. Vous pouvez ajouter un maximum de cinq balises aux AWS Systems Manager automatisations. Vous pouvez ajouter des balises aux automatisations lors de leur introduction depuis la console ou depuis la ligne de commande, ou après leur exécution à l'aide de la ligne de commande.

Ajout de balises aux automatisations (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation de gauche, sélectionnez Automation (Automatisation).
3. Choisissez le runbook Automation que vous souhaitez exécuter.
4. Sélectionnez Execute automation (Exécuter l'automatisation).

5. Dans la section Balises choisissez Modifier, puis ajoutez une ou plusieurs paires de balises clé-valeur.
6. Choisissez Enregistrer.

Ajout de balises aux automatisations (ligne de commande)

Exécutez la commande suivante à l'aide de l'outil de ligne de commande de votre choix pour ajouter des balises à une automatisation au démarrage. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

Linux & macOS

```
aws ssm start-automation-execution \
    --document-name DocumentName \
    --parameters ParametersRequiredByDocument \
    --tags "Key=ExampleKey,Value=ExampleValue"
```

Windows

```
aws ssm start-automation-execution ^
    --document-name DocumentName ^
    --parameters ParametersRequiredByDocument ^
    --tags "Key=ExampleKey,Value=ExampleValue"
```

PowerShell

```
$exampleTag = New-Object Amazon.SimpleSystemsManagement.Model.Tag
$exampleTag.Key = "ExampleKey"
$exampleTag.Value = "ExampleValue"

Start-SSMAutomationExecution `
    -DocumentName DocumentName `
    -Parameter ParametersRequiredByDocument
    -Tag $exampleTag
```

1. Vous pouvez également vous servir de l'outil de ligne de commande de votre choix pour baliser les automatisations après leur exécution. Exécutez la commande suivante pour ajouter des balises à une automatisation. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

Linux & macOS

```
aws ssm add-tags-to-resource \
    --resource-type "Automation" \
    --resource-id "automation-execution-id" \
    --tags "Key=ExampleKey,Value=ExampleValue"
```

Windows

```
aws ssm add-tags-to-resource ^
    --resource-type "Automation" ^
    --resource-id "automation-execution-id" ^
    --tags "Key=ExampleKey,Value=ExampleValue"
```

PowerShell

```
$exampleTag = New-Object Amazon.SimpleSystemsManagement.Model.Tag
$exampleTag.Key = "ExampleKey"
$exampleTag.Value = "ExampleValue"

Add-SSMResourceTag `
    -ResourceType "Automation" `
    -ResourceId "automation-execution-id" `
    -Tag $exampleTag `
    -Force
```

En cas de réussite, la commande n'a aucune sortie.

2. Exécutez la commande suivante pour vérifier les balises de l'automatisation.

Linux & macOS

```
aws ssm list-tags-for-resource \
    --resource-type "Automation" \
    --resource-id "automation-execution-id"
```

Windows

```
aws ssm list-tags-for-resource ^
    --resource-type "Automation" ^
```

```
    --resource-id "automation-execution-id"
```

PowerShell

```
Get-SSMResourceTag `
    -ResourceType "Automation" `
    -ResourceId "automation-execution-id"
```

Suppression de balises des automatisations

Vous pouvez utiliser un outil de ligne de commande pour supprimer des balises d'une automatisations.

Suppression de balises des automatisations (ligne de commande)

1. Exécutez la commande suivante à l'aide de l'outil de ligne de commande de votre choix pour retirer une balises d'une automatisation. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   Linux & macOS

   ```
   aws ssm remove-tags-from-resource \
       --resource-type "Automation" \
       --resource-id "automation-execution-id" \
       --tag-key "tag-key"
   ```

   Windows

   ```
   aws ssm remove-tags-from-resource ^
       --resource-type "Automation" ^
       --resource-id "automation-execution-id" ^
       --tag-key "tag-key"
   ```

   PowerShell

   ```
   Remove-SSMResourceTag `
       -ResourceId "automation-execution-id" `
       -ResourceType "Automation" `
       -TagKey "tag-key" `
   ```

```
    -Force
```

2. Exécutez la commande suivante pour vérifier les balises de l'automatisation.

   Linux & macOS

   ```
   aws ssm list-tags-for-resource \
       --resource-type "Automation" \
       --resource-id "automation-execution-id"
   ```

   Windows

   ```
   aws ssm list-tags-for-resource ^
       --resource-type "Automation" ^
       --resource-id "automation-execution-id"
   ```

   PowerShell

   ```
   Get-SSMResourceTag `
       -ResourceType "Automation" `
       -ResourceId "automation-execution-id"
   ```

Balisage des documents Systems Manager

Les rubriques de cette section décrivent l'utilisation de balises sur les documents Systems Manager (documents SSM).

Rubriques

- Création de documents avec des balises
- Ajout de balises à des documents existants
- Suppression de balises des documents SSM

Création de documents avec des balises

Vous pouvez ajouter des balises à des documents SSM personnalisés au moment de leur création.

Pour plus d'informations, consultez les rubriques suivantes :

- Créer un document SSM (console)
- Créer un document SSM (ligne de commande)

Ajout de balises à des documents existants

Vous pouvez ajouter des balises aux documents SSM 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)
- Ajout de balises à un document SSM existant (ligne de commande)

Ajout de balises à un document SSM existant (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.
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 Enregistrer.

Ajout de balises à un document SSM existant (ligne de commande)

Pour ajouter des balises à un document SSM 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 & macOS

   ```
   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. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

Linux & macOS

```
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
```

En cas de réussite, la commande n'a aucune sortie.

3. Exécutez la commande suivante pour vérifier les balises du document :

   Linux & macOS

   ```
   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 SSM

Vous pouvez utiliser la console Systems Manager ou la ligne de commande pour supprimer des balises des documents SSM.

Rubriques

- Suppression de balises des documents SSM (console)
- Suppression de balises des documents SSM (ligne de commande)

Suppression de balises des documents SSM (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, cliquez sur Documents.
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 Enregistrer.

Suppression de balises des documents SSM (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 & macOS

   ```
   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. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   Linux & macOS

   ```
   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
```

En cas de réussite, la commande n'a aucune sortie.

3. Exécutez la commande suivante pour vérifier les balises du document :

Linux & macOS

```
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

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)
- Didacticiel : 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 AWS Systems Manager ou de la ligne de commande.

Rubriques

Ajout de balises à une fenêtre de maintenance existante (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Maintenance Windows.
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 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. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations.

   Linux & macOS

   ```
   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.
3. Exécutez la commande suivante pour vérifier les balises de la fenêtre de maintenance.

   Linux & macOS

   ```
   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 (AWS Tools for 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 est 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 voulez 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)
- Suppression de balises des fenêtres de maintenance (ligne de commande)

Suppression de balises des fenêtres de maintenance (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Maintenance Windows.
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 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 & macOS

   ```
   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. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

Linux & macOS

```
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
```

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 & macOS

```
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 nœuds gérés

Les rubriques de cette section décrivent l'utilisation de balises sur les nœuds gérés.

Un nœud géré est une machine configurée pour AWS Systems Manager. Cela comprend les instances Amazon Elastic Compute Cloud (Amazon EC2), ainsi que les machines non EC2 d'un environnement hybride et multicloud configurées pour Systems Manager.

Les instructions de cette rubrique s'appliquent à toute machine gérée à l'aide de Systems Manager.

Rubriques

- Création ou activation de nœuds gérés avec des balises
- Ajout de balises à des nœuds gérés existants
- Suppression des balises des nœuds gérés

Création ou activation de nœuds gérés 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, consultez la section Marquer vos ressources Amazon EC2 dans le guide de l'utilisateur Amazon EC2. (Le contenu s'applique aux instances EC2 pour Linux et pour Windows)
- Pour les serveurs locaux et les machines virtuelles, voir Créer une activation hybride pour enregistrer des nœuds auprès de Systems Manager.

Ajout de balises à des nœuds gérés existants

Vous pouvez ajouter des balises aux nœuds gérés à l'aide de la console Systems Manager ou de la ligne de commande.

Rubriques

- Ajout de balises à un nœud géré existant (console)
- Ajout de balises à un nœud géré existant (ligne de commande)

Ajout de balises à un nœud géré existant (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Choisissez l'ID du nœud géré auquel ajouter des balises, puis choisissez l'onglet Tags (Balises).

> **Note**
>
> Si, contrairement à vos attentes, un nœud géré ne figure pas dans la liste, consultez Résolution des problèmes de disponibilité des nœuds gérés 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 Enregistrer.

Ajout de balises à un nœud géré existant (ligne de commande)

Pour ajouter des balises à un nœud géré 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 nœuds gérés que vous pouvez baliser.

   Linux & macOS

   ```
   aws ssm describe-instance-information
   ```

   Windows

   ```
   aws ssm describe-instance-information
   ```

   PowerShell

   ```
   Get-SSMInstanceInformation
   ```

   Notez l'ID d'un nœud géré que vous souhaitez baliser.

   > **Note**
   >
   > Les machines non EC2 enregistrées pour une utilisation avec Systems Manager dans un environnement hybride et multicloud commencent par `mi-`, par exemple, `mi-0471e04240EXAMPLE`. Les ID des instances EC2 commencent par `i-`, par exemple `i-02573cafcfEXAMPLE`.

2. Exécutez la commande suivante pour baliser un nœud géré. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations.

Linux & macOS

```
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
```

En cas de réussite, la commande n'a aucune sortie.

3. Exécutez la commande suivante pour vérifier les balises de nœud géré.

Linux & macOS

```
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 des balises des nœuds gérés

Vous pouvez utiliser la console Systems Manager ou la ligne de commande pour supprimer les balises des nœuds gérés.

Rubriques

- Suppression des balises des nœuds gérés (console)
- Suppression de balises des nœuds gérés (ligne de commande)

Suppression des balises des nœuds gérés (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Fleet Manager.
3. Choisissez le nom du nœud géré duquel supprimer les balises, puis choisissez l'onglet Tags (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 Enregistrer.

Suppression de balises des nœuds gérés (ligne de commande)

1. À l'aide de votre outil de ligne de commande préféré, exécutez la commande suivante pour répertorier les nœuds gérés de votre compte.

   Linux & macOS

   ```
   aws ssm describe-instance-information
   ```

   Windows

   ```
   aws ssm describe-instance-information
   ```

   PowerShell

   ```
   Get-SSMInstanceInformation
   ```

   Notez le nom d'un nœud géré dont vous souhaitez supprimer les balises.

2. Exécutez la commande suivante pour supprimer des balises d'un nœud géré. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   Linux & macOS

   ```
   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
```

En cas de réussite, la commande n'a aucune sortie.

3. Exécutez la commande suivante pour vérifier les balises de nœud géré.

Linux & macOS

```
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"
```

Balisage d'OpsItems

Les rubriques de cette section décrivent l'utilisation de balises sur OpsItems.

Rubriques

- Suppression de balises à partir d'OpsItems Systems Manager

Création d'OpsItems avec des balises

Vous pouvez ajouter des balises à des AWS Systems Manager OpsItems personnalisés au moment de leur création si vous utilisez un outil de ligne de commande.

Pour de plus amples informations, veuillez consulter la rubrique suivante :

Ajout de balises à des OpsItems existants

Vous pouvez ajouter des balises à des OpsItems à l'aide d'un outil de ligne de commande.

Rubriques

- Ajout de balises à un OpsItem existant (ligne de commande)

Ajout de balises à un OpsItem existant (ligne de commande)

Pour ajouter des balises à un OpsItem 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 OpsItem que vous pouvez baliser.

   Linux & macOS

   ```
   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. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   Linux & macOS

   ```
   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
   ```

   En cas de réussite, la commande n'a aucune sortie.

3. Exécutez la commande suivante pour vérifier les balises de l'OpsItem.

   Linux & macOS

   ```
   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 à partir d'OpsItems Systems Manager

Vous pouvez utiliser un outil de ligne de commande pour supprimer les balises Systems Manager OpsItems.

Rubriques

- Suppression de balises à partir d'OpsItems (ligne de commande)

Suppression de balises à partir d'OpsItems (ligne de commande)

1. À l'aide de votre outil de ligne de commande préféré, exécutez la commande suivante pour répertorier les OpsItems de votre compte.

   Linux & macOS

   ```
   aws ssm describe-ops-items
   ```

   Windows

   ```
   aws ssm describe-ops-items
   ```

PowerShell

```
Get-SSMOpsItemSummary
```

Notez le nom d'un OpsItem dont vous souhaitez supprimer les balises.

2. Exécutez la commande suivante pour supprimer les identifications d'un OpsItem. Remplacez chaque *example resource placeholder* (espace réservé pour l'exemple de ressource) par vos propres informations.

Linux & macOS

```
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
```

En cas de réussite, la commande n'a aucune sortie.

3. Exécutez la commande suivante pour vérifier les balises de l'OpsItem.

Linux & macOS

```
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 de paramètres Systems Manager

Les rubriques de cette section décrivent comment utiliser les balises sur AWS Systems Manager les paramètres (paramètres SSM).

Rubriques

- Création de paramètres avec des balises
- Ajout de balises aux paramètres existants
- Suppression de balises des paramètres SSM

Création de paramètres avec des balises

Vous pouvez ajouter des balises aux paramètres SSM au moment de leur création.

Pour plus d'informations, consultez les rubriques suivantes :

- Créer un paramètre Systems Manager (console)
- Créer un paramètre Systems Manager (AWS CLI)
- Créer un paramètre Systems Manager (Tools for Windows PowerShell)

Ajout de balises aux paramètres existants

Vous pouvez ajouter des balises aux paramètres SSM 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)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez 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 Enregistrer.

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. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations.

   ```
   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.

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 (AWS Tools for 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. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

```
$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
```

3. Exécutez la commande suivante pour vérifier les balises du paramètre.

```
Get-SSMResourceTag `
    -ResourceType "Parameter" `
    -ResourceId "parameter-name"
```

Suppression de balises des paramètres SSM

Vous pouvez utiliser la console Systems Manager ou la ligne de commande pour supprimer des balises des paramètres SSM.

Rubriques

Suppression de balises des paramètres SSM (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez 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 Enregistrer.

Suppression de balises des paramètres SSM (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 & macOS

   ```
   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. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   Linux & macOS

   ```
   aws ssm remove-tags-from-resource \
       --resource-type "Parameter" \
       --resource-id "parameter-name" \
       --tag-key "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
   ```

   En cas de réussite, la commande n'a aucune sortie.

3. Exécutez la commande suivante pour vérifier les balises du document :

   Linux & macOS

   ```
   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
- Ajout de balises aux références de correctifs existantes
- Suppression de balises des références de correctifs

Création de références de correctifs avec des balises

Vous pouvez ajouter des balises aux lignes de base des AWS Systems Manager correctifs au moment de leur création.

Pour plus d'informations, consultez les rubriques suivantes :

- Utilisation des référentiels de correctifs personnalisés
- Créer un référentiel de correctifs
- Création d'un référentiel de correctifs avec des référentiels personnalisés pour les différentes versions du système d'exploitation

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 un référentiel de correctifs existante (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Patch Manager.
3. Choisissez le nom d'un référentiel de correctifs personnalisée que vous avez déjà créée, faites défiler jusqu'à la section Tableau des balises, puis choisissez Modification des balises.
4. Choisissez 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 Enregistrer les modifications.

Ajout de balises à un référentiel de correctifs existant (AWS CLI)

1. À l'aide de votre outil de ligne de commande préféré, exécutez la commande suivante pour afficher la liste des référentiels de correctifs que vous pouvez baliser.

   ```
   aws ssm describe-patch-baselines --filters "Key=OWNER,Values=[Self]"
   ```

   Notez l'ID d'un référentiel de correctifs que vous souhaitez baliser.
2. Exécutez la commande suivante pour attribuer une balise à un référentiel de correctifs. Remplacez chaque *`example resource placeholder`* (espace réservé pour les ressources) avec vos propres informations.

Linux & macOS

```
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.

3. Exécutez la commande suivante pour vérifier les balises du référentiel de correctifs.

Linux & macOS

```
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 un référentiel de correctifs (AWS Tools for 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 un référentiel de correctifs. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

```
$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
```

3. Exécutez la commande suivante pour vérifier les balises du référentiel 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 du référentiel de correctifs.

Rubriques

- Suppression de balises du référentiels de correctifs (console)
- Suppression de balises des références de correctifs (ligne de commande)

Suppression de balises du référentiels de correctifs (console)

1. Ouvrez la AWS Systems Manager console à l'adresse https://console.aws.amazon.com/systems-manager/.
2. Dans le panneau de navigation, sélectionnez Patch Manager.

3. Choisissez le nom du référentiel 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 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 & macOS

   ```
   aws ssm describe-patch-baselines
   ```

   Windows

   ```
   aws ssm describe-patch-baselines
   ```

   PowerShell

   ```
   Get-SSMPatchBaseline
   ```

   Notez l'ID d'un référentiel de correctifs dont vous souhaitez supprimer des balises.
2. Exécutez la commande suivante pour supprimer des balises d'un référentiel de correctifs. Remplacez chaque *example resource placeholder* (espace réservé pour les ressources) avec vos propres informations.

   Linux & macOS

   ```
   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
```

En cas de réussite, la commande n'a aucune sortie.

3. Exécutez la commande suivante pour vérifier les balises du référentiel de correctifs.

Linux & macOS

```
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"
```

AWS Systems Manager référence

Les informations et les rubriques suivantes peuvent vous aider à mieux mettre en œuvre des solutions AWS Systems Manager .

Principal

Dans AWS Identity and Access Management (IAM), vous pouvez accorder ou refuser à un service l'accès aux ressources à l'aide de l'élément de politique principal. La valeur de l'élément de politique Principal pour Systems Manager est `ssm.amazonaws.com`.

Points de terminaison Régions AWS et terminaux pris en charge

Veuillez consulter la rubrique Points de terminaison de service Systems Manager de la Référence générale d'Amazon Web Services.

Service Quotas

Veuillez consulter la rubrique Quotas de service Systems Manager de la Référence générale d'Amazon Web Services.

API Reference

Consultez Référence des API AWS Systems Manager.

AWS CLI Référence de commande

Voir AWS Systems Manager la section du manuel de référence des AWS CLI commandes.

AWS Tools for PowerShell Référence de l'applet de commande

Consultez AWS Systems Manager la section de référence des AWS Tools for PowerShell applets de commande.

SSM AgentRéférentiel sur GitHub

Voir aws/ amazon-ssm-agent.

Poser une question

Problèmes Systems ManagerAWS re:Post

AWS Blog d'actualités

Outils de gestion

Autres rubriques de référence

- Référence : modèles et types d'événements Amazon EventBridge pour Systems Manager
- Référence : Expressions Cron et Rate pour Systems Manager
- Référence : ec2messages, ssmmessages et autres opérations d'API
- Référence : Création de chaînes de date et d'heure formatées pour Systems Manager

Référence : modèles et types d'événements Amazon EventBridge pour Systems Manager

> **Note**
>
> Amazon EventBridge est la meilleure façon de gérer vos événements. CloudWatch Events et EventBridge représentent les mêmes service et API sous-jacents, mais EventBridge offre davantage de fonctionnalités. Les modifications que vous apportez dans CloudWatch ou EventBridge apparaîtront dans chaque console. Pour plus d'informations, consultez le Guide de l'utilisateur Amazon EventBridge.

Avec Amazon EventBridge, vous pouvez créer des règles correspondant à des événements entrants, et les acheminer vers des cibles à des fins de traitement.

Un événement indique une modification d'un environnement dans vos propres applications, des applications SaaS (Software as a Service) ou un Service AWS. Les événements sont générés sur la base du meilleur effort. Une fois qu'un type d'événement spécifié dans une règle est détecté, EventBridge l'achemine vers une cible spécifiée à des fins de traitement. Les cibles peuvent inclure des instances Amazon Elastic Compute Cloud (Amazon EC2), des fonctions AWS Lambda, des flux Amazon Kinesis, des tâches Amazon Elastic Container Service (Amazon ECS), des machines d'état AWS Step Functions, 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 de règles EventBridge, consultez les rubriques suivantes :

- Surveillance d'événements Systems Manager avec Amazon EventBridge
- Exemples EventBridge d'événements Amazon pour Systems Manager

- Démarrage avec Amazon EventBridge dans le Guide de l'utilisateur Amazon EventBridge.

Le reste de cette rubrique décrit les types d'événements Systems Manager que vous pouvez inclure dans vos règles EventBridge.

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 de l'automatisation EC2	Changement du statut global d'un flux de travail Automation. Vous pouvez ajouter un ou plusieurs des changements de statut suivants à une règle d'événement : • Approuvé • Annulé • Échec • PendingApproval • PendingChangeCalendarOverride • Refusée • Planifié • Réussite • TimedOut
Notification de changement de statut d'une étape d'automatisation EC2	Le statut d'une étape spécifique d'un flux de travail Automation change. Vous pouvez ajouter un ou plusieurs des changements de statut suivants à une règle d'événement : • Annulé • Échec • Réussite • TimedOut

Type d'événement : Change Calendar

Nom du type d'événement	Description des événements que vous pouvez ajouter à une règle
Changements d'état de calendrier	L'état d'un Change Calendar change. Vous pouvez ajouter un des changements de statut suivants, ou les deux, à une règle d'événement : • OPEN • CLOSED Les modifications d'état pour les planifications partagées à partir d'autres Comptes AWS ne sont pas prises en charge.

Type d'événement : Change Manager

Nom du type d'événement	Description des événements que vous pouvez ajouter à une règle
Mise à jour du statut des demandes de modification	L'état d'une demande de modification de Change Manager. Vous pouvez utiliser les états suivants dans une règle d'événement : • Approuvé • Refusée • InProgress

Type d'événement : conformité de configuration

Nom du type d'événement	Description des événements que vous pouvez ajouter à une règle
Changements d'état de la conformité de configuration	L'état d'un nœud géré change, en termes de conformité des associations ou de conformité des correctifs. Vous pouvez ajouter un ou plusieurs des changements d'état suivants à une règle d'événements : • conforme • non_compliant

Type d'événement : Inventory

Nom du type d'événement	Description des événements que vous pouvez ajouter à une règle
Changements d'état de la ressource Inventory	La suppression d'un inventaire personnalisé et d'un appel PutInventory utilisant une ancienne version de schéma. Vous pouvez ajouter un ou plusieurs des changements d'état suivants à une règle d'événements : • Événement de type d'inventaire personnalisé supprimé sur un nœud spécifique. EventBridge envoie un événement par nœud pour chaque InventoryType personnalisé. • Événement de type inventaire personnalisé supprimé sur tous les nœuds. • Appel putInventory avec un événement utilisant une ancienne version de schéma. EventBridge envoie cet événement lorsque la version du schéma est inférieure au schéma

Nom du type d'événement	Description des événements que vous pouvez ajouter à une règle
	actuel. Cet événement s'applique à tous les types d'inventaire. Pour de plus amples informations, veuillez consulter À propos de la surveillance d'événements Inventory par EventBridge.

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 plusieurs fenêtres de maintenance change. Vous pouvez ajouter un ou plusieurs des changements d'état suivants à une règle d'événements : • DISABLED • ENABLED
Notification d'enregistrement de cible de fenêtre de maintenance	Le statut d'une ou plusieurs cibles de fenêtre de maintenance change. Vous pouvez ajouter un ou plusieurs des changements d'état suivants à une règle d'événements : • DEREGISTERED • REGISTERED • MIS À JOUR
Notification de changement d'état d'exécution de fenêtre de maintenance	Le statut global d'une fenêtre de maintenance change pendant qu'elle s'exécute. Vous pouvez ajouter un ou plusieurs des changements d'état suivants à une règle d'événements :

Nom du type d'événement	Description des événements que vous pouvez ajouter à une règle
	• CANCELLED • CANCELLING • ÉCHEC • IN_PROGRESS • PENDING • SKIPPED_OVERLAPPING • SUCCESS • TIMED_OUT
Notification de changement d'état de l'exécution d'une tâche de fenêtre de maintenance	Le statut d'une tâche de fenêtre de maintenance change pendant qu'elle s'exécute. Vous pouvez ajouter un ou plusieurs des changements d'état suivants à une règle d'événements : • CANCELLED • CANCELLING • ÉCHEC • IN_PROGRESS • SUCCESS • TIMED_OUT

Nom du type d'événement	Description des événements que vous pouvez ajouter à une règle
Notification de changement de statut d'une invocation de cible de la tâche de la fenêtre de maintenance	Le statut d'une tâche de fenêtre de maintenan ce sur une cible spécifique change. Cette notification n'est prise entièrement en charge que pour les tâches Run Command. Pour ce type de tâche, vous pouvez ajouter un ou plusieurs des changements d'état suivants à une règle d'événements : • CANCELLED • CANCELLING • ÉCHEC • IN_PROGRESS • SUCCESS • TIMED_OUT Pour Automation, AWS Lambda et les tâches AWS Step Functions, EventBridge indique uniquement les états `IN_PROGRESS` et `COMPLETE`. `COMPLETE` est indiqué si la tâche a abouti ou non.
Notification d'enregistrement de tâche de fenêtre de maintenance	L'état d'une ou plusieurs tâches de fenêtre de maintenance change. Vous pouvez ajouter un ou plusieurs des changements d'état suivants à une règle d'événements : • DEREGISTERED • REGISTERED • MIS À JOUR

Type d'événement : OpsCenter

Nom du type d'événement	Description des événements que vous pouvez ajouter à une règle
Création d'OpsItem	Se produit lors de la création d'OpsItem. Vous pouvez créer des règles pour l'un des types d'OpsItem suivants : • /aws/issue • /aws/task • /aws/insight • /aws/actionitem
Mise à jour de OpsItem	Se produit lorsqu'un OpsItem est mis à jour. Vous pouvez créer des règles pour l'un des types d'OpsItem suivants : • /aws/issue • /aws/task • /aws/insight • /aws/actionitem

Type d'événement : Parameter Store

Nom du type d'événement	Description des événements que vous pouvez ajouter à une règle
Changement de Parameter Store	L'état d'un paramètre change. Vous pouvez ajouter un ou plusieurs des changements d'état suivants à une règle d'événements : • Création • Mettre à jour • Supprimer

Nom du type d'événement	Description des événements que vous pouvez ajouter à une règle
	• LabelParameterVersion Pour de plus amples informations, veuillez consulter Configuration des EventBridge règles pour les paramètres et des politiques de paramètres.
Action de politique Parameter Store	Une condition d'un changement de politique de paramètres avancée est remplie. Vous pouvez ajouter un ou plusieurs des changements de statut suivants à une règle d'événement : • Expiration • ExpirationNotification • NoChangeNotification Pour de plus amples informations, veuillez consulter Configuration des EventBridge règles pour les paramètres et des politiques de paramètres.

Type d'événement : Run Command

Nom du type d'événement	Description des événements que vous pouvez ajouter à une règle
Notification de changement de statut d'une invocation de commande EC2	Le statut d'une commande envoyée à une instance gérée individuelle change. Vous pouvez ajouter un ou plusieurs des changements de statut suivants à une règle d'événement : • Réussite • InProgress

Nom du type d'événement	Description des événements que vous pouvez ajouter à une règle
	• TimedOut • Annulé • Échec
Notification de changement de statut EC2 Command	Le statut global d'une commande change. Vous pouvez ajouter un ou plusieurs des changements de statut suivants à une règle d'événement : • Réussite • InProgress • TimedOut • Annulé • Échec

Type d'événement : State Manager

Nom du type d'événement	Description des événements que vous pouvez ajouter à une règle
Changement d'état d'association EC2 State Manager	L'état global d'une association change à mesure qu'elle est appliquée. Vous pouvez ajouter un ou plusieurs des changements d'état suivants à une règle d'événements : • Échec • En suspens • Réussite
Changements d'état d'association d'instance EC2 State Manager	L'état d'une instance gérée unique ciblée par une association change. Vous pouvez ajouter un ou plusieurs des changements d'état suivants à une règle d'événements :

Nom du type d'événement	Description des événements que vous pouvez ajouter à une règle
	• Échec • En suspens • Réussite

Référence : Expressions Cron et Rate pour Systems Manager

Lorsque vous créez une association State Manager ou une fenêtre de maintenance dans AWS Systems Manager, vous spécifiez un calendrier d'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.

Informations générales sur les expressions cron et rate

Les informations suivantes s'appliquent aux expressions cron et rate pour les fenêtres de maintenance et les associations.

Planification en un seul cycle

Lorsque vous créez une association ou 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. Par exemple : `"at(2020-07-07T15:55:00)"`

Décalages de planification

Les associations et les fenêtres de maintenance prennent 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 l'association ou la fenêtre de maintenance.

Maintenance window example

Dans la commande suivante, l'expression cron planifie une fenêtre de maintenance à exécuter le troisième mardi de chaque mois à 23 h 30. Cependant, en raison du décalage horaire qui est 2, la fenêtre de maintenance ne s'exécutera qu'à 23 h 30 deux jours plus tard.

```
aws ssm create-maintenance-window \
    --name "My-Cron-Offset-Maintenance-Window" \
```

```
    --allow-unassociated-targets \
    --schedule "cron(30 23 ? * TUE#3 *)" \
    --duration 4 \
    --cutoff 1 \
    --schedule-offset 2
```

Association example

Dans la commande suivante, l'expression cron planifie l'exécution d'une association le deuxième jeudi de chaque mois. Cependant, en raison du décalage horaire qui est 3, l'association ne fonctionnera que le dimanche suivant, trois jours plus tard.

```
aws ssm create-association \
    --name "AWS-UpdateSSMAgent" \
    --targets "Key=instanceids,Values=i-0cb2b964d3e14fd9f" \
    --schedule-expression "cron(0 0 ? * THU#2 *)"  \
    --schedule-offset 3
    --apply-only-at-cron-interval
```

> **Note**
>
> Pour utiliser un décalage avec une association, vous devez spécifier l'option `--apply-only-at-cron-interval`. Cette option indique au système de ne pas exécuter d'association immédiatement après sa création.

Si vous créez une association ou une fenêtre de maintenance avec une expression cron qui cible un jour qui est déjà passé dans la période actuelle, mais que vous ajoutez une date de décalage de planification qui tombe dans le futur, l'association ou la fenêtre de maintenance ne s'exécutera 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.

Champs obligatoires

Les expressions cron pour les fenêtres de maintenance comportent six champs obligatoires. Les expressions Cron pour les associations en ont cinq. (State Manager ne prend actuellement pas en charge la spécification de mois dans les expressions cron pour les associations.) Un champ supplémentaire, 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	18	?	*	MON-FRI	*	Exécuter à 18 h 00 (UTC) du lundi au vendredi
0	8	1	*	?	*	Exécuter à 8 h 00 (UTC) chaque 1er jour du mois

Valeurs prises en charge

Le tableau suivant montre les valeurs prises en charge pour les entrées cron obligatoires.

Valeurs prises en charge pour les expressions cron

Champ	Valeurs	Caractères génériques
Minutes	`0-59`	`, - * /`

Champ	Valeurs	Caractères génériques
Heures	`0-23`	`, - * /`
D ay-of-month	`1-31`	`, - * ? / L W`
Mois (fenêtres de maintenan ce uniquement)	`1-12` ou `JAN-DEC`	`, - * /`
D ay-of-week	`1-7` ou `SUN-SAT`	`, - * ? / L #`
Année	`1970-2199`	`, - * /`

> **Note**
>
> Vous ne pouvez pas spécifier de valeur dans les champs day-of-month et dans les day-of-week champs d'une 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.

Caractères génériques pour les expressions cron

Le tableau suivant présente les valeurs de caractères génériques que les expressions cron prennent en charge.

> **Note**
>
> Les expressions cron qui entraînent des fréquences d'une rapidité supérieure à cinq (5) minutes ne sont pas prises en charge. Support pour spécifier à la fois un day-of-week et une day-of-month valeur n'est pas complet. Utilisez un point d'interrogation « ? » dans l'un de ces champs.

Caractères génériques pris en charge pour les expressions cron

Caractère générique	Description
,	Le caractère générique `,` , (virgule) inclut des valeurs supplémentaires. Dans le champ

Caractère générique	Description
	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'interro gation) indique l'un ou l'autre. Dans le ay-of-month champ D, vous pouvez saisir 7 et si vous ne vous souciez pas du jour de la semaine le 7, vous pouvez entrer ? dans le ay-of-week champ D.
L	Le `L` caractère générique dans les ay-of-week champs D ay-of-month ou D indique le dernier jour du mois ou de la semaine.
W	Le `W` caractère générique dans le ay-of-month champ D indique un jour de la semaine. Dans le ay-of-month champ D, 3W indique le jour le plus proche du troisième jour de semaine du mois.

Caractère générique	Description
#	Le # caractère générique dans le day-of-week champ suivi d'un chiffre compris entre un et cinq indique un jour donné du mois. 5 #3 indique le 3e jeudi du mois.

Expressions de fréquence

Les expressions de fréquence comportent deux champs obligatoires. Ces champs sont séparés par un espace.

Champs obligatoires pour les expression rate

Champ	Valeurs
Valeur	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.

Rubriques

- Expressions cron et rate pour les associations
- Expressions cron et rate pour les fenêtres de maintenance

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 informations suivantes :

- Les associations prennent en charge les expressions cron suivantes : toutes les 1/2, 1, 2, 4, 8 ou 12 heures ; chaque jour, chaque semaine, ou chaque jour et heure précisés de la semaine ; un jour précis dans une semaine précise du mois ou le dernier jour x du mois à une heure précise.
- Les associations prennent en charge les expressions rate suivantes : intervalles de 30 minutes ou plus et moins de 31 jours.
- Si vous spécifiez le champ facultatif `Seconds`, sa valeur peut être 0 (zéro). Par exemple : `cron(0 */30 * * * ? *)`
- Pour une association qui collecte des métadonnées pour Inventory, une des fonctionnalités de AWS Systems Manager, nous vous recommandons d'utiliser une expression rate.
- State Manager ne prend actuellement pas en charge la spécification de mois dans les expressions cron pour les associations.

Les associations prennent en charge les expressions cron qui incluent un jour de la semaine et le signe numérique (#) pour désigner le nième jour d'un mois pour diriger une association. Voici un exemple qui exécute une planification cron le troisième mardi de chaque mois à 23 h 30 UTC :

```
cron(30 23 ? * TUE#3 *)
```

Voici un exemple qui se déroule le deuxième jeudi de chaque mois à minuit UTC :

```
cron(0 0 ? * THU#2 *)
```

Les associations soutiennent également le signe (L) pour indiquer le dernier X jour du mois. Voici un exemple qui exécute une planification cron le dernier mardi de chaque mois à 23 h 30 UTC :

```
cron(0 0 ? * 3L *)
```

Pour contrôler davantage l'exécution d'une association, par exemple si vous souhaitez exécuter une association deux jours après le correctif mardi, vous pouvez spécifier un décalage. Un offset (décalage) définit le nombre de jours d'attente après le jour prévu pour exécuter une association. Par exemple, si vous avez spécifié une planification cron de `cron(0 0 ? * THU#2 *)`, vous pouvez spécifier le numéro 3 dans le champ Schedule offset (Décalage de planification) pour exécuter l'association tous les dimanches après le deuxième jeudi du mois.

Pour utiliser des décalages, vous devez choisir le Apply association only at the next specified Cron interval (Appliquer l'association uniquement à l'intervalle Cron spécifié suivant) dans la console ou vous devez spécifier l'utilisation du paramètre `--apply-only-at-cron-interval` à partir de la ligne de commande. Cette option indique à State Manager ne pas exécuter d'association immédiatement après sa création.

Le tableau suivant présente des exemples d'expressions cron pour les associations.

Exemples cron pour associations

Exemple	Détails
cron(0/30 * * * ? *)	Toutes les 30 minutes
cron(0 0/1 * * ? *)	Toutes les heures
cron(0 0/2 * * ? *)	Toutes les 2 heures
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
cron(30 23 ? * TUE#3 *)	Le troisième mardi de chaque mois à 23 h 30

Voici quelques exemples d'expressions rate pour les associations.

Exemples rate pour les associations

Exemple	Détails
rate(30 minutes)	Toutes les 30 minutes
rate(1 hour)	Toutes les heures
rate(5 hours)	Toutes les 5 heures

Exemple	Détails
rate(15 days)	Tous les 15 jours

Exemples de AWS CLI pour les 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 la AWS CLI sur une machine Linux locale.

> **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. Spécifiez `--apply-only-at-cron-interval` de sorte que l'association ne s'exécute pas immédiatement après sa création. Ce paramètre n'est pas pris en charge pour les expressions rate.

```
aws ssm create-association \
    --association-name "My-Cron-Association" \
    --schedule-expression "cron(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
```

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.

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 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	Détails
cron(0 2 ? * THU#3 *)	2 h 00 le troisième jeudi de chaque mois
cron(15 10 ? * * *)	10 h 15 tous les jours
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	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

Exemples de AWS CLI pour les fenêtres de maintenance

Pour créer des fenêtres de maintenant à l'aide de l'AWS CLI, vous incluez le paramètre `--schedule` avec une expression cron ou rate, ou un horodatage. Les exemples suivants utilisent la 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-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
```

Plus d'informations

Expression CRON sur le site web Wikipedia

Référence : ec2messages, ssmmessages et autres opérations d'API

Si vous surveillez les opérations d'API, vous pouvez voir des appels aux opérations suivantes :

- `ec2messages:AcknowledgeMessage`
- `ec2messages:DeleteMessage`
- `ec2messages:FailMessage`
- `ec2messages:GetEndpoint`
- `ec2messages:GetMessages`
- `ec2messages:SendReply`
- `ssmmessages:CreateControlChannel`
- `ssmmessages:CreateDataChannel`
- `ssmmessages:OpenControlChannel`
- `ssmmessages:OpenDataChannel`
- `ssm:DescribeDocumentParameters`
- `ssm:DescribeInstanceProperties`
- `ssm:GetCalendar`
- `ssm:GetManifest`
- `ssm:ListInstanceAssociations`
- `ssm:PutCalendar`
- `ssm:PutConfigurePackageResult`
- `ssm:RegisterManagedInstance`
- `ssm:RequestManagedInstanceRoleToken`

- `ssm:UpdateInstanceAssociationStatus`
- `ssm:UpdateInstanceInformation`
- `ssm:UpdateManagedInstancePublicKey`

Il s'agit d'opérations spéciales utilisées par AWS Systems Manager, comme décrit dans le reste de cette rubrique.

Opérations d'API liées à l'agent (**`ssmmessages`**et **`ec2messages`** points de terminaison)

Opérations d'API ssmmessages

Systems Manager utilise le `ssmmessages` point de terminaison pour les deux types d'opérations d'API suivants :

- Des opérations de SSM Agent àSession Manager, une capacité de AWS Systems 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. De plus, si la connectivité est autorisée, SSM Agent reçoit `Command` des documents par ce biaisAmazon Message Gateway Service. Si la connectivité n'est pas autorisée, SSM Agent reçoit `Command` les documents via leAmazon Message Delivery Service. Pour de plus amples informations, consultez Actions, ressources et clés de condition pour Amazon Session Manager Message Gateway Service
- Opérations depuis Systems Manager Agent (SSM Agent) jusqu'au service Systems Manager dans le cloud.

Opérations d'API ec2messages

`ec2messages:*`Les opérations d'API sont effectuées sur le point de terminaison Amazon Message Delivery Service. Systems Manager utilise ce point de terminaison pour effectuer des opérations d'API, de Systems Manager Agent (SSM Agent) au service Systems Manager dans le cloud.

> **Important**
>
> `ec2messages:*`Les opérations d'API ne sont prises en charge Régions AWS que dans celles lancées avant 2024. Dans les régions lancées à partir de 2024, seules les opérations `ssmmessages:*` d'API sont prises en charge.

Priorité de connexion au point de terminaison

À partir de la version 3.3.40.0 de, Systems SSM Agent Manager a commencé à utiliser le `ssmmessages:*` point de terminaison (Amazon Message Gateway Service) chaque fois qu'il était disponible au lieu du `ec2messages:*` point de terminaison (). Amazon Message Delivery Service

Si vous fournissez l'accès `ssmmessages:*` à vos politiques d'autorisation AWS Identity and Access Management (IAM), vous vous connectez au point SSM Agent de `ssmmessages:*` terminaison, même si votre profil d'instance IAM est configuré pour autoriser les deux points de terminaison. Cela inclut les politiques pour les profils d'instance IAM et les rôles de service IAM que vous avez créés vous-même, ainsi que pour les profils d'instance IAM créés par la configuration de gestion d'Quick Setuphôte et la configuration de gestion d'hôte par défaut.

Si vous avez fourni des autorisations pour les deux points de terminaison et que vous surveillez les opérations d'API à l'aide, par exemple, de CloudWatch Metrics, vous ne verrez aucun appel à`ec2messages:*`.

Pour les Régions AWS versions lancées avant 2024 : vous pouvez supprimer les `ec2messages:*` autorisations de vos politiques en toute sécurité pour le moment.

Basculement de la connexion du terminal

Pour les Régions AWS applications lancées avant 2024 uniquement : si votre profil d'instance IAM ne fournit pas d'autorisations `ssmmessages:*` au moment du démarrage de l'agent, mais SSM Agent se connecte uniquement `ec2messages:*` au `ec2messages:*` point de terminaison. Si vous avez les deux `ssmmessages:*` et que vous les supprimez `ec2messages:*` au SSM Agent démarrage de l'`ssmmessages:*`agent, bascule SSM Agent rapidement la connexion vers le `ec2messages:*` point de terminaison. Pour les régions lancées à partir de 2024, seul le `ssmmessages:*` point de terminaison est pris en charge.

Pour plus d'informations sur les `ec2messages:*` points de terminaison `ssmmessages` et, consultez les rubriques suivantes dans la référence d'autorisation AWS de service.

- Actions, ressources et clés de condition pour Amazon Message Gateway Service (`ssmmessages`).
- Actions, ressources et clés de condition pour Amazon Message Delivery Service (`ec2messages:*`)

ssm:*opérations d'API liées à l'instance d'espace de noms

`DescribeDocumentParameters`

Systems Manager exécute cette opération d'API pour afficher des nœuds spécifiques dans la console Amazon EC2. Les résultats de l'`DescribeDocumentParameters`opération sont affichés dans le nœud Documents.

`DescribeInstanceProperties`

Systems Manager exécute ces opérations d'API pour afficher des nœuds spécifiques dans la console Amazon EC2. Les résultats de l'opération `DescribeInstanceProperties` sont affichés dans le nœud Fleet Manager.

`GetCalendar`

Systems Manager exécute cette opération d'API pour afficher les documents Change Calendar de type dans la Change Calendar console.

`GetManifest`

SSM Agentexécute cette opération d'API pour déterminer la configuration système requise pour l'installation ou la mise à jour d'une version spécifiée d'un AWS Systems Manager Distributor package. Il s'agit d'une ancienne opération d'API qui n'est pas disponible si elle Régions AWS a été lancée après 2017.

`ListInstanceAssociations`

SSM Agentexécute cette opération d'API pour voir si une nouvelle State Manager association est disponible. Cette opération d'API est requise pour que State Manager fonctionne.

`PutCalendar`

Systems Manager exécute cette opération d'API pour mettre à jour les documents Change Calendar de type dans la Change Calendar console.

`PutConfigurePackageResult`

SSM Agentexécute cette opération d'API pour publier les mesures d'erreur d'installation et de latence des packages de distribution publics sur le compte du propriétaire du package.

`RegisterManagedInstance`

SSM Agentexécute cette opération d'API pour les scénarios suivants :

- Enregistrer un serveur local ou une machine virtuelle (VM) auprès de Systems Manager en tant qu'instance gérée à l'aide d'un code et d'un ID d'activation.
- Pour enregistrer les AWS IoT Greengrass Version 2 informations d'identification.

Cette opération est également appelée par les instances Amazon EC2 exécutant la version 3.1.x ou ultérieure de l'SSM Agent.

`RequestManagedInstanceRoleToken`

SSM Agentexécute cette opération d'API pour récupérer les informations d'identification temporaires permettant d'accéder au nœud géré.

`UpdateInstanceAssociationStatus`

SSM Agentexécute cette opération d'API pour mettre à jour une association. Cette opération d'API est requise pour State Manager qu'une capacité de AWS Systems Manager, fonctionne.

`UpdateInstanceInformation`

SSM Agentappelle le service Systems Manager dans le cloud toutes les 5 minutes pour fournir des informations sur le rythme cardiaque. Cet appel est nécessaire pour maintenir une pulsation avec l'agent afin que le service sache que l'agent fonctionne comme prévu.

`UpdateManagedInstancePublicKey`

SSM Agentexécute cette opération d'API pour fournir la clé publique après avoir fait pivoter la paire de clés sur le nœud géré. La clé publique est utilisée pour authentifier les demandes, signées avec la clé privée, afin d'obtenir des informations d'identification temporaires auprès de Systems Manager.

Référence : Création de chaînes de date et d'heure formatées pour Systems Manager

Les opérations d'API AWS Systems Manager acceptent des filtres pour limiter le nombre de résultats renvoyés par une demande. Certaines de ces opérations d'API acceptent des filtres qui nécessitent une chaîne formatée pour représenter une date et une heure spécifiques. Par exemple, l'opération d'API `DescribeSessions` accepte les clés `InvokedAfter` et `InvokedBefore` parmi les valeurs valides pour un objet `SessionFilter`. L'opération 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, consultez ISO 8601.

Ces chaînes de date et d'heure formatées ne sont pas limitées aux filtres. Il existe également des opérations 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'opération `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 opérations 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 opérations 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 AWS Tools for PowerShell et l'AWS Command Line Interface (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
```

AWS Tools for 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 meilleures pratiques en matière AWS Systems Manager de fonctionnalités. 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

- Créez des runbooks Automation en libre-service pour l'infrastructure.
- Utilisez Automation, une fonctionnalité permettant de AWS Systems Manager simplifier la création Amazon Machine Images (AMIs) à partir AWS Marketplace ou sur mesureAMIs, à l'aide de documents publics de Systems Manager (documents SSM) ou en créant vos propres flux de travail.
- Générez et gérez des AMIs à l'aide des runbooks Automation `AWS-UpdateLinuxAmi` et `AWS-UpdateWindowsAmi`, ou à l'aide de runbooks Automation personnalisés que vous créez.

Inventaire

- Utilisez Inventory, une fonctionnalité de AWS Systems Manager, AWS Config pour auditer les configurations de vos applications au fil du temps.

Maintenance Windows

- Définissez un calendrier pour la mise en œuvre des actions potentiellement perturbatrices, comme l'application de correctifs à un système d'exploitation, la mise à jour de pilotes ou l'installation de logiciels.
- Pour plus d'informations sur les différences entre State Manager etMaintenance Windows, les fonctionnalités de AWS Systems Manager, voirChoisir entre State Manager et Maintenance Windows.

Parameter Store

- UtilisezParameter Store, une fonctionnalité de AWS Systems Manager, pour gérer de manière centralisée les paramètres de configuration globaux.
- Comment AWS Systems ManagerParameter Store utilise AWS KMS.
- Référencez AWS Secrets Manager les secrets à partir Parameter Store des paramètres.

Patch Manager

- Utilisez Patch Manager une fonctionnalité pour déployer des AWS Systems Manager correctifs à grande échelle et améliorer la visibilité de la conformité de la flotte sur l'ensemble de vos nœuds.
- Intégrez Patch Manager à AWS Security Hub pour recevoir des alertes lorsque des nœuds de votre flotte ne sont plus conformes, et pour contrôler l'état d'application de correctifs de vos flottes du point de vue de la sécurité. L'utilisation de Security Hub entraîne des frais supplémentaires. Pour plus d'informations, consultez Pricing (Tarification CTlong).
- Utilisez une seule méthode à la fois pour analyser la conformité aux correctifs des nœuds gérés, afin d'éviter de remplacer involontairement les données de conformité.

Run Command

- Gérer les instances à grande échelle sans accès SSH en utilisant la fonctionnalité Exécuter la commande d'EC2.
- Auditez tous les appels d'API effectués par ou au nom d'Run Commandune fonctionnalité AWS Systems Manager d'utilisation de AWS CloudTrail.
- Lorsque vous envoyez une commande à l'aide de Run Command, n'incluez pas d'informations sensibles formatées en texte brut, comme des mots de passe, des données de configuration ou d'autres secrets. Toutes les activités de l'API Systems Manager sur votre compte sont enregistrées dans un compartiment S3 pour les AWS CloudTrail journaux. Cela signifie que tout utilisateur ayant accès à ce compartiment S3 peut consulter les valeurs en texte brut de ces secrets. Pour cette raison, nous vous recommandons de créer et d'utiliser des paramètres `SecureString` pour chiffrer les données sensibles que vous utilisez dans le cadre de vos opérations Systems Manager.

  Pour plus d'informations, consultez Restriction de l'accès aux paramètres Systems Manager à l'aide des politiques IAM.

> **Note**
>
> Par défaut, les fichiers journaux envoyés par CloudTrail votre compartiment sont chiffrés par chiffrement côté serveur Amazon avec des clés de chiffrement gérées par Amazon S3 (SSE-S3). Pour fournir une couche de sécurité directement gérable, vous pouvez plutôt utiliser le chiffrement côté serveur avec des clés AWS KMS gérées (SSE-KMS) pour vos fichiers journaux. CloudTrail
> Pour plus d'informations, consultez la section Chiffrement des fichiers CloudTrail journaux à l'aide de AWS KMS clés gérées (SSE-KMS) dans le guide de l'utilisateur.AWS CloudTrail

- Utilisez les cibles et les fonctions de contrôle du débit de Run Command pour exécuter des commandes par étapes.
- Utilisez des autorisations d'accès détaillées pour Run Command (et toutes les fonctionnalités de Systems Manager) en utilisant des politiques AWS Identity and Access Management (IAM).

Session Manager

- Auditez l'activité des sessions que vous Compte AWS utilisez AWS CloudTrail.
- Enregistrez les données de session dans votre compte à Compte AWS l'aide d'Amazon CloudWatch Logs ou d'Amazon S3.
- Contrôlez l'accès des sessions utilisateur aux instances.
- Restreindre l'accès aux commandes dans une session.
- Désactivez ou activez les autorisations administratives du compte ssm-user.

State Manager

- Mettez à jour l'SSM Agent au moins une fois par mois à l'aide du document préconfiguré `AWS-UpdateSSMAgent`.
- (Windows) Téléchargez le module PowerShell ou DSC sur Amazon Simple Storage Service (Amazon S3) et utilisez-le. `AWS-InstallPowerShellModule`
- Utilisez des balises afin de créer des groupes d'applications pour vos nœuds. Puis, ciblez des nœuds en utilisant le paramètre `Targets` au lieu de spécifier des ID de nœud individuels.
- Corrigez automatiquement les résultats générés par Amazon Inspector en utilisant Systems Manager.

- Utilisez un référentiel de configuration centralisé pour vos documents SSM, et partagez les documents dans l'ensemble de votre organisation.
- Pour plus d'informations sur les différences entre State Manager et Maintenance Windows, consultez Choisir entre State Manager et Maintenance Windows.

Nœuds gérés

- Systems Manager nécessite des références horaires précises pour effectuer ses opérations. Si la date et l'heure de votre nœud ne sont pas correctement définies, elles risquent de ne pas correspondre à la date de signature de vos demandes d'API. Cela peut conduire à des erreurs ou des fonctionnalités incomplètes. Par exemple, les nœuds dont les paramètres horaires sont incorrects ne seront pas inclus dans vos listes de nœuds gérés.

  Pour plus d'informations sur le réglage de l'heure sur vos nœuds, consultez Définir l'heure pour votre instance Amazon EC2.
- Sur les nœuds gérés par Linux, vérifiez la signature de SSM Agent.

Plus d'informations

- Bonnes pratiques de sécurité pour Systems Manager

Suppression de ressources et d'artefacts Systems Manager

Une bonne pratique consiste à supprimer les ressources et les artefacts Systems Manager si vous n'avez plus besoin d'afficher les données relatives à ces ressources ou d'utiliser les artefacts de quelque manière que ce soit. Le tableau suivant répertorie chaque fonctionnalité ou artefact Systems Manager, ainsi qu'un lien vers des informations supplémentaires sur la suppression des ressources ou des artefacts créés par Systems Manager.

Fonctionnalité ou artefact	Détails
Application Manager	Vous ne pouvez pas supprimer une application dans Application Manager, mais vous pouvez la supprimer du service en supprimant les balises, groupes de ressources ou piles AWS CloudFormation.

Fonctionnalité ou artefact	Détails
Automatisation	Si vous créez des AWS ressources à l'aide de Systems Manager Automation, vous devez les supprimer manuellement à l'aide de la commande correspondante AWS Management Console. Si vous avez créé un runbook personnalisé, vous pouvez supprimer le document SSM sous-jacent. Pour plus d'informations, consultez Suppression de documents SSM personnalisés.
Change Calendar	Vous pouvez supprimer un calendrier de modification et un événement de calendrier de modification. Pour plus d'informations, consultez Suppression d'un calendrier de modifications et Suppression d'un événement Change Calendar.
Change Manager	Vous pouvez supprimer un modèle de modification. Pour plus d'informations, consultez Suppression de modèles de modification.
Conformité d'	Systems Manager Compliance affiche automatiquement les données de conformité sur l'application de correctifs Patch Manager et les associations State Manager. Vous ne pouvez pas supprimer ces données. Si vous avez configuré une synchronisation des données de ressources pour centraliser les données de conformité dans un compartiment S3, vous pouvez supprimer la synchronisation. Pour plus d'informations, consultez Suppression d'une synchronisation de données de ressources pour le service Conformité.

Fonctionnalité ou artefact	Détails
Distributor	Vous pouvez supprimer des packages dans Distributor. Pour plus d'informations, consultez Supprimer un package.
Explorer	Vous pouvez vous déconnecter des sources à partir desquelles Explorer les données sont collectées OpsData. Pour plus d'informations, consultez Modification de sources de données Systems Manager Explorer. Vous pouvez également supprimer une synchronisation des données de ressource s utilisée Explorer pour agréger OpsData et OpsItems entre plusieurs comptes Régions AWS et vers un seul compartiment Amazon Simple Storage Service (Amazon S3). Pour plus d'informations, consultez Suppression des données de ressource Systems Manager Explorer. Pour plus d'informations sur la suppression d'un compartiment S3, consultez Suppression d'un compartiment dans le Guide du développeur Amazon Simple Email Service.
Fleet Manager	Vous ne pouvez pas supprimer un nœud géré à l'aide de Fleet Manager. Vous devez utiliser Amazon Elastic Compute Cloud (Amazon EC2). Pour de plus amples informations, consultez Terminate your instance (Linux) (Résilier votre instance (Linux)) et Terminate your instance (Windows) (Résilier votre instance (Windows)).

Fonctionnalité ou artefact	Détails
Inventory	Vous pouvez arrêter la collecte de données d'inventaire en supprimant les associations State Manager qui définissent la planification et les ressources à partir desquelles collecter les métadonnées. Pour plus d'informations, consultez Arrêt de la collecte des données et suppression des données d'inventaire. Si vous ne souhaitez plus utiliser l' AWS Systems Manager inventaire pour afficher les métadonnées relatives à vos AWS ressource s, nous vous recommandons également de supprimer les synchronisations des données de ressources utilisées pour la collecte des données d'inventaire. Pour plus d'informations, consultez Suppression d'une synchronisation de données de ressources Inventory.
Maintenance Windows	Vous pouvez supprimer une fenêtre de maintenance, une cible de fenêtre de maintenance et une tâche de fenêtre de maintenance. Pour plus d'informations, consultez Mise à jour ou suppression de ressources de fenêtre de maintenance (console).
OpsCenter	Vous pouvez supprimer un individu OpsItem en appelant l'opération Delete OpsItem API à l'aide du SDK AWS Command Line Interface ou du AWS SDK. Vous ne pouvez pas supprimer un OpsItem dans la AWS Management Console. Pour plus d'informations, consultez Supprimez OpsItems.

Fonctionnalité ou artefact	Détails
Parameter Store	Vous pouvez supprimer un paramètre que vous avez créé. Pour plus d'informations, consultez Suppression de paramètres Systems Manager.
Patch Manager	Vous pouvez supprimer un référentiel de correctifs personnalisé. Pour plus d'informations, consultez Mise à jour ou suppression d'un référentiel de correctifs personnalisé.
Configuration rapide	Vous pouvez supprimer les associations créées par la fonctionnalité Configuration rapide. Les associations sont stockées et traitées par State Manager. Pour plus d'informations, consultez Suppression d'associations.
Run Command	Après qu'une commande a terminé un traitement, les informations correspondantes sont stockées sous l'onglet Command history (Historique des commandes). Vous ne pouvez pas supprimer les informations de l'onglet Historique des commandes.
Rôle lié à un service	Systems Manager crée automatiquement des rôles liés au service pour certaines fonctionnalités. Vous pouvez supprimer ces rôles. Pour plus d'informations, consultez Suppression du rôle lié au service `AWSServiceRoleForAmazonSSM` pour Systems Manager.
Session Manager	Session Manager ne conserve pas les données relatives à vos ressources au terme d'une session. Pour terminer une session, consultez Résilier une session.

Fonctionnalité ou artefact	Détails
SSM Agent	Vous pouvez désinstaller manuellement SSM Agent à partir de vos nœuds. Pour plus d'informations, consultez les rubriques suivantes. • Linux : Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour Linux • macOS: Installation et désinstallation manuelles SSM Agent sur les instances EC2 pour macOS • Windows Server : ouvrez le panneau de commande et sélectionnez Add/remove programs (Ajouter/supprimer des programmes).
State Manager	Vous pouvez supprimer une association. Pour plus d'informations, consultez Suppression d'associations.
Service de document Systems Manager	Vous ne pouvez pas supprimer les runbooks fournis par AWS ou AWS Support, mais vous pouvez supprimer les runbooks personnalisés. Pour plus d'informations, voir Suppression de documents SSM personnalisés.

Choisir entre State Manager et Maintenance Windows

State ManageretMaintenance Windows, les deux fonctionnalités de AWS Systems Manager, peuvent effectuer des types de mises à jour similaires sur vos nœuds gérés. Votre choix dépend de la nécessité d'automatiser la conformité du système ou d'effectuer des tâches hautement prioritaires et sensibles au temps pendant les périodes que vous spécifiez.

State Manager et Maintenance Windows : cas d'utilisation clés

State Manager, une fonctionnalité permettant de AWS Systems Manager définir et de maintenir la configuration d'état ciblée pour les nœuds et les AWS ressources gérés au sein de votre Compte AWS. Vous pouvez définir des combinaisons de configurations et de cibles en tant qu'objets d'association. State Manager est la fonctionnalité recommandée pour maintenir tous les nœuds gérés de votre compte dans un état cohérent, utiliser Amazon EC2 Auto Scaling pour générer de nouveaux nœuds ou avoir des exigences strictes en matière de rapports de conformité pour les nœuds gérés de votre compte.

Les principaux cas d'utilisation pour State Manager sont les suivants :

- Scénarios Auto Scaling : State Manager peut contrôler tous les nouveaux nœuds lancés dans un compte, soit manuellement, soit via des groupes Auto Scaling. Si le compte comporte des associations qui ciblent ce nouveau nœud (par le biais de balises ou de tous les nœuds), cette association particulière est automatiquement appliquée au nouveau nœud.
- Rapports de conformité : State Manager peut générer des rapports de conformité des états souhaités des ressources de votre compte.
- Prise en charge de tous les nœuds : State Manager peut cibler tous les nœuds d'un compte donné.

Une fenêtre de maintenance effectue une ou plusieurs actions sur des ressources AWS à l'intérieur d'une fenêtre de temps donnée. Vous pouvez définir une seule fenêtre de maintenance avec les heures de début et de fin. Vous pouvez spécifier plusieurs tâches à exécuter à l'intérieur de cette fenêtre de maintenance. Utilisez une fonctionnalité permettant Maintenance Windows AWS Systems Manager, si vos opérations prioritaires incluent l'application de correctifs à vos nœuds gérés, l'exécution de plusieurs types de tâches sur vos nœuds pendant une période de mise à jour ou le contrôle du moment où les opérations de mise à jour peuvent être exécutées sur vos nœuds.

Les principaux cas d'utilisation pour Maintenance Windows 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 à l'intérieur d'une seule fenêtre de maintenance.
- Application de correctifs : une fenêtre de maintenance peut prévoir la prise en charge de l'application de correctifs pour tous les nœuds gérés d'une seule région balisés avec une balise

ou un groupe de ressources spécifique. Comme le processus d'application de correctifs implique généralement la mise hors service des nœuds (par exemple, le retrait des nœuds d'un équilibreur de charge), l'application des correctifs et le post-traitement (remise en production des nœuds), il peut prendre la forme d'une série de tâches réalisées dans une fenêtre de temps donnée.

> **Note**
>
> Lorsque vous utilisez une fenêtre de maintenance, votre opération d'application de correctifs est limitée à une seule région dans un seul compte. À l'aide d'une politique de correctifs créée dans Quick Setup, une fonctionnalité de Systems Manager, vous pouvez configurer l'application de correctifs pour certains ou tous les comptes et régions d'une organisation créée dans AWS Organizations. Pour plus d'informations, consultez Utilisation des stratégies de correctifs Quick Setup.

- Actions de la fenêtre : les fenêtres de maintenance peuvent faire en sorte qu'un ou plusieurs ensembles d'actions démarrent à l'intérieur d'une fenêtre de temps spécifique. Les fenêtres de maintenance ne permettront pas le démarrage en dehors de cette fenêtre. Les actions déjà commencées se poursuivent jusqu'à leur terme, même si elles se terminent en dehors de la fenêtre de temps.

Le tableau suivant compare les principales fonctions de State Manager et des Maintenance Windows.

Fonctionnalité	State Manager	Maintenance Windows
AWS CloudFormation intégration	AWS CloudFormation les modèles soutiennent State Manager les associations.	AWS CloudFormation les modèles prennent en charge les fenêtres de maintenance, les cibles de fenêtres et les tâches liées aux fenêtres.
Conformité	Chaque association State Manager signale la conformité par rapport au statut souhaité de la ressource ciblée. Vous pouvez utiliser le tableau de bord de conformité pour	Ne s'applique pas.

Fonctionnalité	State Manager	Maintenance Windows
	agréger et afficher la conformité signalée.	
Intégration de la gestion de la configuration	State Managerprend en charge les solutions d'état ciblées externes telles que Microsoft PowerShell Desired State Configuration (DSC), les Ansible playbooks et Chef les recettes. Vous pouvez utiliser les associations State Manager pour tester le fonctionnement des solutions de gestion de la configuration et pour appliquer les changements de configuration à vos nœuds lorsque vous êtes prêt.	Ne s'applique pas.
Documents	Des configurations State Manager peuvent être définies en tant que documents de politique (pour la collecte d'informations d'inventaire), runbooks Automation, pour des ressources AWS telles que les compartiments Amazon Simple Storage Service (Amazon S3), ou documents de commande Systems Manager (documents SSM) pour les nœuds gérés.	Des configurations Maintenance Windows peuvent être définies en tant que documents d'automatisation (actions en plusieurs étapes avec des flux d'approbation facultatifs) ou documents SSM (état souhaité pour les nœuds gérés).

Fonctionnalité	State Manager	Maintenance Windows
Surveillance	State Manager surveille les changements de configuration, d'association ou d'état d'un nœud (par exemple, la mise en ligne de nouveaux nœuds). Lorsque State Manager détecte ces changemen ts, l'association donnée est réappliquée aux nœuds initialement ciblés par celle-ci.	Ne s'applique pas.
Priorités appliquées aux tâches	Ne s'applique pas.	Une priorité peut être affectée aux tâches à l'intérieur d'une fenêtre de maintenance. Toutes les tâches de même priorité sont exécutées en parallèle. Les tâches de priorité inférieure sont exécutées une fois que les tâches de priorité élevée ont atteint un état final. Il n'existe aucun moyen d'exécuter des tâches sous certaines conditions. Une fois qu'une tâche de priorité supérieure atteint son état final, la tâche de priorité suivante s'exécute , indépendamment de l'état de la tâche précédente.

Fonctionnalité	State Manager	Maintenance Windows
Commandes de sécurité	State Manager prend en charge deux contrôles de sécurité lors du déploiement de configurations sur une flotte étendue. Vous pouvez utiliser la concomitance maximale pour définir le nombre de nœuds ou de ressources sur lesquels la configuration doit être simultanément appliquée . Vous pouvez définir un taux d'erreur maximal pour la suspension de l'association State Manager si un certain nombre ou pourcentage d'erreurs se produit au sein de la flotte.	Les fenêtres de maintenance prennent en charge deux contrôles de sécurité lors du déploiement de configurations sur une flotte étendue. Vous pouvez utiliser la concomitance maximale pour définir le nombre de nœuds ou de ressources sur lesquels la configuration doit être simultanément appliquée. Vous pouvez définir un taux d'erreur maximal pour la suspension des actions dans une fenêtre de maintenance si un certain nombre ou pourcentage d'erreurs se produit sur la flotte.

Fonctionnalité	State Manager	Maintenance Windows
Planification	Vous pouvez exécuter des associations State Manager à la demande, à un intervall e cron particulier, à un débit donné, ou une fois leur création. Cela est utile pour maintenir l'état souhaité de vos ressources de manière cohérente et opportune. **Important** Les expressions cron pour les associati ons State Manager ne prennent pas en charge le champ des mois, tel que `03` ou `MAR` pour le mois de mars. Si vous avez besoin de mises à jour de configura tion mensuelles ou trimestrielles, une fenêtre de maintenan ce peut répondre au mieux à vos besoins. Pour plus d'informa tions, consultez Référence : Expressio ns Cron et Rate pour Systems Manager.	Les fenêtres de maintenance prennent en charge plusieurs options de planification, notamment des expressions `at` (par exemple, `"at(2021-07-07T13:15:30)"`), des expressions cron et rate, des cron avec décalages, des heures de début et de fin définissant les périodes durant lesquelles les fenêtres de maintenance doivent s'exécuter, et des heures de coupure pour spécifier à quel moment arrêter la planification à l'intérieur d'une fenêtre de temps donnée.

Fonctionnalité	State Manager	Maintenance Windows
Ciblage	Les associations State Manager peuvent cibler un ou plusieurs nœuds à l'aide d'un ID de nœud, d'une balise ou d'un groupe de ressources. State Manager peut cibler tous les nœuds gérés d'un compte donné.	Les fenêtres de maintenance peuvent cibler un ou plusieurs nœuds à l'aide d'ID de nœud, de balises ou de groupes de ressources.

Fonctionnalité	State Manager	Maintenance Windows
Tâches à l'intérieur de 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 du runbook Automation ou du document de commande. Toutes les tâches d'une fenêtre de maintenance s'exécutent en parallèle, sauf si des priorités différentes sont définies pour différentes tâches. Dans l'ensemble, les fenêtres de maintenance prennent en charge quatre types de tâches : • Commandes de l'AWS Systems Manager Run Command • AWS Systems Manager Flux de travail d'automat isation • AWS Lambda fonctions • AWS Step Functions tâches

Informations connexes

Les ressources connexes suivantes peuvent s'avérer utiles lors de l'utilisation de ce service.

Tarification

Certaines fonctionnalités Systems Manager font l'objet d'une facturation. Pour en savoir plus, consultez AWS Systems Manager Tarification.

Bibliothèque de documentation AWS Systems Manager

Documentation AWS Systems Manager : accédez à toute la documentation utilisateur relative à Systems Manager, y compris AWS AppConfig, Incident Manager et AWS Systems Manager pour SAP.

AWS re:Post

AWS re:Post : un service AWS géré de questions et réponses (Q&A) proposant des réponses participatives et révisées par des experts à vos questions techniques.

Blog et podcast AWS

Lisez des articles de blog sur Systems Manager dans la Catégorie d'outils de gestion AWS, et d'autres messages étiquetés avec `#Systems Manager`.

Quotas de service

Veuillez consulter la rubrique Quotas de service Systems Manager de la Référence générale d'Amazon Web Services. Sauf indication contraire, chaque quota s'applique à une seule région d'un Compte AWS.

Référence d'autorisation de service pour Systems Manager

Dans la Référence d'autorisation de service AWS, consultez des informations sur les actions, les ressources et les clés de contexte de condition que vous pouvez utiliser dans les politiques AWS Identity and Access Management (IAM) pour Systems Manager.

Contrat de niveau de service AWS Systems Manager

Le contrat de niveau de service (SLA) AWS Systems Manager est une politique régissant l'utilisation de Systems Manager. Elle s'applique séparément à chaque Compte AWS utilisant Systems Manager.

Ressources AWS générales

Les ressources générales suivantes peuvent s'avérer utiles lorsque vous travaillez avez AWS.

- 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.
- Centre pour développeurs AWS : parcourez des didacticiels, téléchargez des outils et découvrez les événements pour les développeurs AWS.
- Outils pour développeur AWS : liens vers des outils pour développeur, kits SDK, boîtes à outils IDE et outils de ligne de commande pour développer et gérer des applications AWS.
- Centre de ressources pour la mise en route : découvrez comment configurer votre Compte AWS, rejoindre la communauté AWS et lancer votre première application.
- Tutoriels pratiques — Suivez les step-by-step didacticiels pour lancer votre première application surAWS.
- 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.
- AWS SupportCentre – Hub pour la création et la gestion de vos cas AWS Support. Inclut également des liens vers d'autres ressources utiles, telles que des forums, des FAQ techniques, l'état de santé d'un service et AWS Trusted Advisor.
- AWS Support— La principale page Web contenant des informations sur AWS Support un one-on-one canal d'assistance à réponse rapide pour vous aider à créer 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.
- AWSConditions d'utilisation du site : informations détaillées sur nos droits d'auteur et notre marque, sur votre compte, votre licence et votre accès au site, et sur d'autres sujets.

Historique du document

Le tableau suivant décrit les modifications importantes apportées à la documentation depuis la dernière version de AWS Systems Manager. Pour recevoir les notifications concernant les mises à jour de cette documentation, abonnez-vous à un flux RSS.

- Version de l'API : 2014-11-06

Modification	Description	Date
Mise à jour : disponibilité régionale du chemin des `/aws/service/global-infrastructure` paramètres	Nous avons précisé à partir de quelles régions commerciales le chemin du paramètre `/aws/service/global-infrastructure` public peut être demandé et comment exécuter une requête pour le chemin si vous travaillez dans une autre publicité. Région AWS Pour plus d'informations, consultez la section Appeler les paramètres publics pour les AWS services, les régions, les points de terminaison, les zones de disponibilité, les zones locales et les zones de longueur d'onde.	12 juin 2024
Nouveau : chapitre sur les exemples de code	Un nouveau chapitre, Code examples for Systems Manager using AWS SDK, fournit des exemples dans différents langages de SDK expliquant comment utiliser le service Systems Manager.	8 mai 2024

Modifications apportées au support des `ec2messages:*` terminaux	Pour un Régions AWS lancement en 2024 ou ultérieurement, les `ec2messages:*` points de terminaison ne sont pas pris en charge SSM Agent pour renvoyer les informations d'état et d'exécution au service Systems Manager. Les comptes de ces régions doivent utiliser `ssmmessages:*` . Dans les régions lancées avant 2024, les deux `ssmmessages:*` `ec2messages:*` sont toujours prises en charge, mais nous recommandons d'utiliser uniquement le `ssmmessages:*` point de terminaison (Amazon Message Gateway Service) pour le moment. Pour le moment, vous pouvez supprimer `ec2messages:*` les autorisations de vos politiques en toute sécurité. Pour plus d'informations, consultez la section Utilisation des SSM Agent opérations d'API associées à l'agent (points de terminaison ssmmessages et ec2messages).	3 mai 2024

Runtimes supplémentaires disponibles pour exécuter des scripts dans les runbooks d'automatisation	L'`aws:executeScript` action prend désormais en charge les environnements d'exécution Python 3.9, 3.10 et 3.11. Pour plus d'informations sur l'utilisation de cette action, consultez `aws:executeScript` .	23 avril 2024
Support pour les versions 8.8 et 8.9 : AlmaLinuxOracle Linux, et Rocky Linux	Systems Manager prend désormais en charge les versions 8.8 et 8.9 de AlmaLinuxOracle Linux, etRocky Linux, en plus des versions 8.x antérieures. Pour obtenir la liste complète des systèmes d'exploitation et des versions pris en charge, voir Systèmes d'exploitation pris en charge pour Systems Manager.	22 avril 2024

Patch Manager: passage au statut du correctif « INSTALLED_PENDING_REBOOT »	Auparavant, seuls les correctifs installés par Patch Manager pouvaient être marqués comme`INSTALLED_PENDING_REBOOT`. Les patchs installés en dehors de n'Patch Manageront jamais reçu ce statut. Il `INSTALLED_PENDING_REBOOT` peut désormais s'appliquer à n'importe quel correctif appliqué à un nœud géré depuis son dernier redémarrage. Cela inclut les correctifs installés Patch Manager avec l'`NoReboot`option sélectionnée, ainsi que les correctifs installés en dehors du Patch Manager dernier redémarrage du nœud. Pour une description de toutes les valeurs d'état d'Patch Managerapplication des correctifs, voir Comprendre les valeurs d'état de conformité des correctifs.	16 avril 2024
Support pour RHEL 8.9 et 9.3	Systems Manager, y compris Patch Manager, prend désormais en charge les versions 8.9 et 9.3 Red Hat Enterprise Linux (RHEL), en plus des versions 8.x et 9.x antérieures.	26 mars 2024

Mise à jour du sujet : politiques AWS gérées pour AWS Systems Manager	La rubrique politiques AWS gérées pour AWS Systems Manager fournit des informations sur les quatre politiques gérées pour Systems Manager qui ont été introduites ou mises à jour depuis le 12 mars 2021. Nous avons ajouté une section à cette rubrique contenant des informations sur les 12 autres politiques gérées à utiliser avec Systems Manager qui ont été créées ou mises à jour pour la dernière fois avant cette date. Pour plus de détails, consultez la section Politiques gérées supplémentaires pour Systems Manager.	1er mars 2024

Parameter Storeprend désormais en charge le partage entre comptes	Vous pouvez désormais partager des paramètres avancés de manière sûre et efficace au sein de votre AWS organisation Comptes AWS ou au sein de celle-ci en configurant le partage des ressources. Le partage des ressources vous permet de centraliser la gestion de la configuration des applicati ons et de réduire les frais opérationnels liés au partage des paramètres avec chaque compte que vous possédez. Les paramètres peuvent être partagés entre les comptes à l'aide de la Parameter Store AWS RAM console, de la console ou du AWS CLI. Pour plus d'informations, consultez la section Utilisation de paramètres partagés.	21 février 2024
Amélioration des actions d'automatisation	Vous pouvez désormais utiliser les `isCritical` propriétés `onFailure` et avec l'`aws:approve` action. Pour plus d'informations sur cette `aws:approve` action, consultez aws:approve — Suspendre une automatisation pour approbation manuelle.	12 février 2024

Support de version d'exploitation supplémentaire pour Patch Manager	Nous avons ajouté à la liste des versions de système d'exploitation prises en charge pour Patch Manager. Support a été ajouté pour les éléments suivants : • Debian Server11.x et 12.x • macOS14,0 (Sonoma) • SUSE Linux Enterprise Server(SLES) 15,5 • Ubuntu Server23,04	4 janvier 2024
Configuration des mises à jour automatisées de SSM Agent à l'aide de la console Application Manager	Vous pouvez désormais utiliser la console Application Manager pour automatiser les mises à jour de SSM Agent pour vos instances d'application. Pour plus d'informations, veuillez consulter Working with your application instances.	21 décembre 2023
Processus mis à jour pour l'enregistrement de machines autres qu'Amazon EC2 dans des environnements hybrides et multicloud	Systems Manager propose désormais l'option `ssm-setup-cli` pour vous aider à enregistrer des machines autres qu'Amazon Elastic Compute Cloud (Amazon EC2) dans des environnements hybrides et multicloud. Pour plus d'informations, consultez Comment installer SSM Agent les nœuds Linux hybrides et Comment installer SSM Agent les nœuds Windows hybrides.	20 décembre 2023

Gestion des volumes Amazon EBS à l'aide de Fleet Manager	Vous pouvez désormais utiliser Fleet Manager une fonctionnalité de AWS Systems Manager pour gérer les volumes Amazon Elastic Block Store sur vos instances gérées. Par exemple, vous pouvez initialiser un volume EBS, formater une partition et monter le volume pour le rendre utilisable. Pour plus d'informations, veuillez consulter EBS volume management.	14 décembre 2023
Amélioration du plugin Session Manager	Ajout de la prise en charge de la transmission d'une réponse d'StartSessionAPI en tant que variable d'environnement à session-manager-plugin.	4 décembre 2023
Nouvelle expérience de conception visuelle pour les runbooks Automation	Vous pouvez désormais créer et modifier des runbooks à l'aide d'une nouvelle expérience de conception visuelle développée par Systems Manager Automation. L'expérience de conception visuelle fournit une drag-and-drop interface à faible code qui vous permet de créer et de modifier des runbooks plus facilement. Pour plus d'informations, consultez Expérience de conception visuelle pour les runbooks d'Automation.	26 novembre 2023

Nouvelles actions de Systems Manager Automation, éléments de données et améliorations fonctionnelles pour les runbooks	Vous pouvez désormais parcourir plusieurs actions dans un runbook à l'aide de l'action `aws:loop`. Cette nouvelle action soutient les boucles de style `do while` et `for each`. En outre, à l'aide du nouvel élément de données variables, vous pouvez définir, référencer et mettre à jour des valeurs de manière dynamique dans le contexte d'un runbook. Pour mettre à jour la valeur d'une variable dans votre runbook, utilisez la nouvelle action `aws:updateVariable`. Automation a également ajouté la prise en charge des conversions dynamiques de types de données pour les sorties. Cela signifie que si la valeur d'une sortie ne correspond pas au type de données que vous avez spécifié, Automation essaie de convertir le type de données. Par exemple, si la valeur renvoyée est `Integer`, mais que le `Type` spécifié est `String`, la valeur de sortie finale est une valeur `String`. Enfin, l'Automatisation prend désormais en charge les expressions de filtre JSONPath pour les sélecteurs. Pour plus d'informa	17 novembre 2023

	tions, consultez les rubriques suivantes : • aws:loop : itérer les étapes dans une automatisation • aws:updateVariable : met à jour la valeur d'une variable runbook • Éléments de données et paramètres : éléments de données de niveau supérieur • Utilisation des sorties d'action comme entrées. • Utilisation de JSONPath dans des runbook.	
Support régional mis à jour pour les connexions Remote Desktop Protocol (RDP)	Le bureau à distance Fleet Manager, qui est alimenté par NICE DCV, vous fournit une connectivité sécurisée à vos instances Windows Server directement depuis la console Systems Manager. Les trois régions supplémentaires suivantes ont été activées pour les connexions du bureau à distance Fleet Manager : • Afrique (Le Cap) (af-south-1) • Asie-Pacifique (Jakarta) (ap-southeast-3) • Israël (Tel Aviv) (il-central-1)	15 novembre 2023

Patch Manager : prise en charge étendue des versions de systèmes d'exploitation pour RHEL et macOS	Patch Manager prend désormais en charge les versions de système d'exploitation supplémentaires suivantes : • Red Hat Enterprise Linux : version 8.8 • macOS : 11.5 à 11.7 (Big Sur) • macOS : 12.0 à 12.6 (Monterey) • macOS : 13.0 à 13.5 (Ventura)	23 octobre 2023
Nouvelle API OpsCenter – DeleteOpsItem	OpsCenter offre désormais l'API DeleteOpsItem pour supprimer des OpsItems individuels. Pour plus d'informations, consultez la section DeleteOpsÉlément de la référence de AWS Systems Manager l'API.	20 octobre 2023

Nouveau type Quick Setup de configuration : SSM Agent mises à jour pour l'ensemble de l'organisation	Le nouveau type de configuration Configuration de gestion d'hôte par défaut permet à un administrateur de l'organisation, tel que défini dans AWS Organizations, de demander la vérification et la mise à jour automatiques de SSM Agent toutes les instances EC2 des comptes et des régions de l'organisation. Pour plus d'informations, consultez Gestion des hôtes par défaut pour une organisation.	16 octobre 2023
Nouveau format de titre et de description OpsItems créé par CloudWatch Application Insights	Le titre et la description de OpsItems Created by CloudWatch Application Insights passeront à un format amélioré le 16 octobre 2023. Pour consulter le nouveau format, consultez Amazon CloudWatch Application Insights.	29 septembre 2023

Prise en charge de plusieurs résolutions d'affichage dans les connexions RDP Fleet Manager	Lorsque vous vous connectez à des nœuds Windows Server gérés à l'aide de l'option RDP (Remote Desktop Protocol) dans Fleet Manager, vous pouvez désormais choisir la résolution d'affichage. Auparavant, toutes les connexions utilisaient une résolution fixe de 720P (1366 x 768). Vous pouvez désormais choisir parmi les options suivantes pour chaque connexion : • Adaptation automatique (détermine la résolution optimale en fonction de la taille de votre écran détectée) • 1920 x 1080 • 1400 x 900 • 1#366 x 768 • 800 x 600 Pour plus d'informations, consultez Connexion à un nœud géré à l'aide du Bureau à distance.	22 septembre 2023

Nouveau rubrique : ID référentiel de correctifs aléatoires dans les opérations relatives aux politique de correctifs	Nous avons ajouté du contenu pour décrire comment les politiques de correctifs Quick Setup utilisent le paramètre `BaselineOverride` du document de commande SSM `AWS-RunPatchBaseline` pour générer des ID aléatoires pour les référentiels de correctifs chaque fois qu'une opération de politique de correctifs est exécutée. Pour plus d'informations, consultez ID référentiel de correctifs aléatoires dans les opérations relatives aux politique de correctifs.	22 septembre 2023

Une nouvelle vision opérationnelle pour la gestion d'OpsItems	OpsCenterinclut désormais un aperçu opérationnel appelé Resources generating the most OpsItems. Un aperçu de ce type est généré lorsqu'une AWS ressource en a plus de 10 ouvertesOpsItems. Utilisez cette information pour localiser les ressources problématiques. Utilisez le runbook `AWS-BulkResolveOpsItems` à partir d'une information pour résoudre rapidement les OpsItems associés à une ressource. Pour plus d'informations, consultez la section Analyse des informations opérationnelles pour réduire OpsItems.	22 septembre 2023
Clé publique GPG mise à jour	Une nouvelle clé publique a été créée pour vérifier la signature de SSM Agent. Pour plus d'informations, consultez Vérification de la signature de SSM Agent.	5 septembre 2023

Support ajouté pour les versions supplémentaires de AlmaLinuxOracle Linux,RHEL, et Rocky Linux	Les listes des systèmes d'exploitation pris en charge pour AWS Systems Manager et Patch Manager ont été mises à jour pour refléter la prise en charge des versions supplémentaires suivantes du système d'exploitation : • AlmaLinux: 9,2 • Oracle Linux : 8.7 et 9.2 • Red Hat Enterprise Linux (RHEL) : 8.7, 9.1 et 9.2 • Rocky Linux : 8.6 et 8.7, 9.0 à 9.2	30 août 2023

OpsCenter a ajouté la prise en charge du formatage Markdown dans le champ de description OpsItem.	OpsCenter prend désormais en charge le formatage Markdown dans le champ de description OpsItem. Les types de formatage Markdown suivants sont pris en charge : • Paragraphes • Espacement des lignes • Lignes horizontales • En-têtes • Mise en forme d'un texte • Liens • Listes Pour plus d'informations, consultez la section Utilisation de Markdown dans la console dans le guide de AWS Management Console démarrage.	18 août 2023

Nouvelles versions de l'extension Lambda AWS Parameters and Secrets	De nouvelles versions de l'extension Lambda AWS Parameters and Secrets sont désormais disponibles. En outre, la prise en charge de l'extension a été ajoutée pour les régions Asie-Pacifique (Melbourne) (ap-southeast-4) et Israël (Tel Aviv) (il-central-1) (architectures x86_64 et x86 uniquement). Pour plus d'informations, consultez la section Utilisation de Parameter Store paramètres dans AWS Lambda les fonctions.	16 août 2023

Mise à jour : ajout d'informations sur les autorisations requises pour les compartiments de politiques de correctifs Quick Setup	Lorsque vous créez une politique de correctifs, Quick Setup crée un compartiment Amazon S3 qui contient un fichier nommé `baseline_overrides.json` . Ce fichier contient des informations sur les référentiels de correctifs que vous avez spécifiées pour votre politique de correctifs. Lors de la configuration de la politique de correctifs, vous avez la possibilité de cocher la case Ajouter les politiques IAM requises aux profils d'instance existants attachés à vos instances. Si vous choisissez de ne pas sélectionner cette option, vous devez fournir manuellement à certaines ressources des autorisations d'accès à ce compartiment, faute de quoi les opérations de votre politique risquent d'échouer. Pour plus d'informations, consultez les rubriques suivantes : • Autorisations pour le compartiment S3 de la politique de correctifs • Problème : erreur « InvokePatchBaselineOperation : accès refusé » ou erreur « Impossible de télécharger	6 juillet 2023

	le fichier depuis S3 » pour `baseline_overrides.json`	
Utilisation de Quick Setup pour configurer OpsCenter pour la gestion de OpsItem sur plusieurs comptes	L'utilisation de Quick Setup pour OpsCenter vous permet d'effectuer les tâches suivantes pour gérer OpsItems sur plusieurs comptes : • Spécification du compte administrateur délégué • Création de politiques et de rôles requis AWS Identity and Access Management (IAM) • Spécification d'une AWS Organizations organisation, ou d'un sous-ensemble de comptes de membres, où un administrateur délégué peut gérer OpsItems plusieurs comptes Pour plus d'informations, consultez (Facultatif) Configurer OpsCenter pour gérer OpsItems sur plusieurs comptes à l'aide de Quick Setup.	19 juin 2023

Mettre à jour les agents de lancement Amazon EC2 en utilisant Quick Setup	Vous pouvez désormais autoriser Systems Manager à vérifier tous les 30 jours si une nouvelle version de l'agent de lancement est installée sur votre instance. Si une nouvelle version est disponible, Systems Manager met à jour l'agent sur votre instance. Pour en savoir plus, consultez Gestion des hôtes Quick Setup.	19 juin 2023
Patch Manager prend désormais en charge Ubuntu Server 22.04 LTS.	Vous pouvez désormais utiliser Patch Manager pour appliquer des correctifs aux nœuds Ubuntu Server 22.04 LTS. Comme les autres versions prises en charge deUbuntu Server, la version 22.04 LTS utilise la ligne de base de `AWS-UbuntuDefaultPatchBaseline` correctifs AWS gérés.	15 mai 2023

Systems Manager prend désormais en charge AlmaLinux, notamment Patch Manager	Vous pouvez désormais utiliser Systems Manager pour gérer les nœuds AlmaLinux 8.3-8.7 ; 9.0-9.1. La plupart des règles applicables à RHEL 8 pour l'application de correctifs s'appliquent également à AlmaLinux. AlmaLinux utilise le nouveau `AWS-DefaultAlmaLinuxPatchBaseline` . Pour plus d'informations, consultez les rubriques suivantes : • Installation manuelle SSM Agent sur les AlmaLinux instances • Sélection des correctifs de sécurité • Installation des correctifs • Comment fonctionnent les règles de base des correctifs sur AlmaLinuxRHEL,, et Rocky Linux.	8 mai 2023
Déploiement de l'agent EC2Launch v2 avec Quick Setup	Vous pouvez désormais déployer l'agent EC2Launch v2 avec Quick Setup. Pour plus d'informations, consulter la rubrique Déploiement Distributor Packages avecQuick Setup.	13 avril 2023

Systems Manager prend désormais en charge Amazon Linux 2023	Systems Manager prend désormais en charge le nouveau type d'instance EC2 Amazon Linux 2023 (AL2023), ainsi que les opérations Patch Manager. La plupart des règles qui s'appliquent à Amazon Linux 2 en matière de correctifs s'appliquent également à Amazon Linux 2023. (Patch Manager continue également de prendre en charge la version préliminaire d'Amazon Linux 2022.) Pour plus d'informations, consultez les rubriques suivantes : • Sélection des correctifs de sécurité • Installation des correctifs • Comment fonctionnent les règles de base des correctifs sur Amazon Linux 1, Amazon Linux 2, Amazon Linux 2022 et Amazon Linux 2023	23 mars 2023

Contenu de configuration révisé pour les instances Amazon EC2	Nous avons révisé le contenu de configuration pour les instances Amazon EC2. Il est désormais recommandé d'utiliser la Configuration de gestion des hôtes par défaut récemment publiée pour les autorisations d'instance. Pour plus d'informations, consultez Configurer les autorisations d'instance requises pour Systems Manager.	15 février 2023
Gestion automatique des instances avec la Configuration de gestion des hôtes par défaut	Vous pouvez désormais gérer automatiquement les instances Amazon EC2 dans une Région AWS entière à l'aide de Systems Manager. Pour de plus amples informations, consultez Gestion de l'enregistreur de configuration.	15 février 2023

Ajoutez des documents SSM à vos favoris	Pour vous aider à trouver les documents SSM fréquemment utilisés, vous pouvez désormais ajouter des documents à vos favoris. Vous pouvez ajouter jusqu'à 20 documents à vos favoris par type de document, par Compte AWS et Région AWS. Vous pouvez choisir, modifier et consulter vos favoris depuis la console Systems Manager, dans Documents. Pour plus d'informations, consultez Ajout de documents à vos favoris.	7 février 2023
Implémentez des contrôles des modifications pour les automatisations à l'aide de Change Calendar	En intégrant Automation àChange Calendar, vous pouvez désormais implémenter des contrôles de modification pour toutes les automatisations de votre Compte AWS. Pour plus d'informations, consultez Implémenter des contrôles des modifications pour les automatisations.	24 janvier 2023

Nouveau flux de travail d'approbation dans Change Manager	Le flux de travail d'approbation Change Manager prend désormais en charge les approbations par niveau au lieu des approbations par ligne. Auparavant, chaque approbateur que vous ajoutiez à un niveau d'approbation devait approuver une demande de modification. Sinon, le niveau n'était pas approuvé. Vous pouvez maintenant spécifier le nombre d'approbations requises pour le niveau et ajouter autant d'approbateurs ou plus. Par exemple, vous pouvez demander trois approbations pour un niveau, mais spécifier jusqu'à cinq approbateurs. Les approbations de trois de ces approbateurs sont suffisantes pour approuver le niveau. Pour plus d'informations, consultez la section À propos des approbations dans vos modèles de modification.	23 janvier 2023

Nouveau : configurez l'application de correctifs pour l'ensemble d'une organisation à l'aide d'une politique de correctifs dans Quick Setup	Grâce à Quick Setup, une fonctionnalité de Systems Manager, vous pouvez désormais créer des politiques de correctif à technologie Patch Manager. Une politique de correctifs définit la planification et le référentiel de correctifs à utiliser lors de l'application automatique de correctifs à vos nœuds gérés. À l'aide d'une configuration de politique de correctifs unique, vous pouvez définir l'application de correctifs pour tous les comptes de toutes les régions de votre organisation, uniquement pour les comptes et les régions de votre choix, ou pour une seule paire compte-région. Pour plus d'informations, consultez les rubriques suivantes. • Utilisation des stratégies de correctifs Quick Setup • Automatiser l'application de correctifs à l'échelle de l'organisation à l'aide d'une politique de correctifs Quick Setup	22 décembre 2022

Application Manager s'intègre à Amazon EC2 pour afficher des informations sur vos instances dans le contexte d'une application.	Application Manager affiche l'état et le statut de l'instance et l'intégrité d'Amazon EC2 Auto Scaling pour une application sélectionnée dans un format graphique. L'onglet Instances inclut également un tableau contenant les informations suivantes pour chaque instance de votre application. • État de l'instance (Pending, Stopping, Running, Stopped [En attente, Arrêt, En cours d'exécution, Arrêtée]) • Statut du ping de SSM Agent • Statut et nom du dernier runbook Systems Manager Automation traité sur l'instance • Nombre d'alarmes Amazon CloudWatch Logs par État. • `ALARM` – La métrique ou l'expression se trouve à l'extérieur du seuil défini. • `OK` – La métrique ou l'expression se trouve dans le seuil défini. • `INSUFFICIENT_DATA` – L'alerte vient de commencer, la métrique n'est pas disponible, ou la quantité de données n'est pas suffisante pour	22 décembre 2022

	permettre à la métrique de déterminer le statut de l'alerte. • Intégrité du groupe Auto Scaling pour les groupes de scalabilité automatique parent et individuel	
Planifiez le démarrage et l'arrêt de vos instances Amazon EC2 à l'aide de Quick Setup	Vous pouvez désormais déployer la solution Planificateur de ressources pour automatiser le démarrage et l'arrêt de vos instances Amazon EC2 à l'aide de Quick Setup. Pour plus d'informations, consultez la rubrique Resource Scheduler (Planificateur de ressources).	19 décembre 2022

OpsCenter prend désormais en charge l'utilisation d'OpsItems entre les comptes	OpsCenter prend en charge l'utilisation d'OpsItems depuis un compte de gestion (compte de gestion AWS Organizations ou compte administrateur délégué Systems Manager) et un compte membre au cours d'une session. Une fois la configuration terminée, les utilisateurs peuvent effectuer les types d'actions suivants : • Créer, visualiser et mettre à jour des OpsItems sur un compte membre • Afficher des informations détaillées sur AWS les ressources spécifiées OpsItems dans un compte membre • Démarrer les runbooks Systems Manager Automation pour résoudre les problèmes liés aux ressources AWS d'un compte membre Pour plus d'informations, consultez la rubrique Configurer OpsCenter pour une utilisation avec OpsItems entre les comptes.	16 novembre 2022

Suivez les détails des demandes de Change Manager modification à l'aide de AWS CloudTrail Lake	Vous pouvez désormais utiliser un magasin de données d'événements dans AWS CloudTrail Lake pour saisir et examiner les détails des demandes de modification introduites Change Manager pour votre organisation ou votre compte. Ces informations incluent des détails vérifiables sur l'identité de l'utilisateur qui a créé la demande de modification, l'adresse IP à partir de laquelle la demande a été faite, l'Régions AWS endroit où les modifications ont été effectuées, les ressources ciblées, etc. Pour plus d'informations, consultez les rubriques Monitoring your change request events (Surveillance de vos événements de demande de modification) et Vérifier les détails, les tâches et les échéances d'une demande de modification.	11 novembre 2022

Contrôles de tâches supplémentaires d'automatisation de Systems Manager à l'aide d' CloudWatch alarmes	Vous pouvez désormais mettre en œuvre un contrôle supplémentaire lors de l'exécution d'automatisations sur plusieurs comptes et régions en utilisant des CloudWatch alarmes. En appliquant une CloudWatch alarme métrique ou composite à une automatisation, vous pouvez contrôler le moment où une automatisation s'arrête en fonction des métriques que vous définissez. Pour plus d'informations sur l'application CloudWatch d'une alarme à une automatisation exécutée sur plusieurs comptes et régions, voir Exécuter une automatisation dans plusieurs régions et comptes (console).	9 novembre 2022

Mise à jour : « Utilisation de Parameter Store paramètres dans AWS Lambda les fonctions »	Nous avons fourni des informations supplémentaires pour vous aider à utiliser l'extension Lambda AWS Parameters and Secrets pour récupérer les valeurs des paramètres et les mettre en cache pour une utilisation future dans les fonctions Lambda. L'utilisation de l'extension Lambda peut réduire vos coûts en diminuant le nombre d'appels d'API vers Parameter Store. Pour plus d'informations, consultez la section Utilisation de Parameter Store paramètres dans AWS Lambda les fonctions.	25 octobre 2022

Contrôles de tâches supplémentaires de Systems Manager à l'aide d' CloudWatch alarmes	Vous pouvez désormais implémenter un contrôle supplémentaire lors de l'exécution d'automatisations et de commandes à l'aide d' CloudWatch alarmes. Une CloudWatch alarme peut également être ajoutée à une automatisation ou à une commande lorsqu'elle est enregistrée avec une tâche d'State Manageras sociation ou de fenêtre de maintenance. En appliquant une CloudWatch alarme composite à une automatisation ou à une commande, vous pouvez contrôler le moment où une automatisation ou une commande s'arrête en fonction de la métrique que vous définissez. Pour plus d'informations sur l'application d'une CloudWatch alarme à une automatisation ou à une commande, consultez les procédures suivantes : • Sélection des correctifs de sécurité • Installation des correctifs • Comment fonctionnent les règles de base des correctifs sur Amazon Linux 1, Amazon Linux 2 et Amazon Linux 2022.	26 septembre 2022

Contrôles de tâches supplémentaires de Systems Manager à l'aide d' CloudWatch alarmes	Vous pouvez désormais implémenter un contrôle supplémentaire lors de l'exécution d'automatisations et de commandes à l'aide d' CloudWatch alarmes. Une CloudWatch alarme peut également être ajoutée à une automatisation ou à une commande lorsqu'elle est enregistrée avec une tâche d'State Manageras sociation ou de fenêtre de maintenance. En appliquant une CloudWatch alarme composite à une automatisation ou à une commande, vous pouvez contrôler le moment où une automatisation ou une commande s'arrête en fonction de la métrique que vous définissez. Pour plus d'informations sur l'application d'une CloudWatch alarme à une automatisation ou à une commande, consultez les procédures suivantes : • Exécution d'une automatisation simple • Exécution des commande à partir de la console • Création d'une association • Affecter des taches à une fenêtre de maintenance	26 septembre 2022

Clarification des exigences du niveau des instances avancés	Selon le rapport de nos clients, nous avons clarifié les scénarios vous obligeant à activer le niveau d'instances avancées dans Configuration du niveau d'instance.	21 septembre 2022
Déployez l' CloudWatch agent Amazon à l'aide de Quick Setup	Vous pouvez désormais déployer l' CloudWatch agent Amazon à l'aide deQuick Setup. Pour plus d'informations, consulter la rubrique Déploiement Distributor Packages avecQuick Setup.	20 septembre 2022

La touche PatchGroup « » est désormais prise en charge pour les groupes de correctifs lorsque les métadonnées de l'instance EC2 sont autorisées	Quand tu autoriser les balises dans les métadonnées d'instance EC2, les clés de balises créées ne doivent contenir aucun espace. Jadis, cela empêchait les clients d'ajouter certaines de leurs instances EC2 à des groupes de correctifs dans Patch Manager parce que la clé de balise `Patch Group` a dû être appliqué aux instances.Patch Manager prend désormais en charge les`Patch Group`(avec un espace) et `PatchGroup` (sans espace) comme clé de balise pour l'identification des instances d'un groupe de correctifs. Les instances EC2 où les balises sont autorisées dans les métadonnées d'instance peuvent désormais être ajoutées aux groupes de correctifs dans Patch Manager. Pour plus d'informations, consultez À propos des groupes de correctifs.	31 août 2022

Nouvelle rubrique : « Calcul des dates de sortie et des mises à jour des packages »	Dans les lignes de base de correctifs gérées par AWS, les nouveaux correctifs sont approuvés automatiquement 7 jours après leur publicati on ou leur mise à jour. Dans les référentiels de correctif s personnalisées crées, spécifiez éventuellement le nombre de jours d'attente après leur publication ou leur mise à jour pour l'approba tion automatique de leur installation. Pour Amazon Linux 1 et Amazon Linux 2, divers facteurs influencent le mode de calcul des dernières dates de sortie et de mise à jour. Afin de vous aider à éviter des résultats inattendu s lors du choix des délais d'approbation automatique, ces facteurs sont expliqués dans la rubrique Calcul des dates de sortie et des mises à jour des packages.	24 août 2022

Contenu mis à jour : Corriger une AMI et mettre à jour un groupe Auto Scaling	Nous avons mis à jour la démonstration Mise à jour des AMIs pour les groupes Auto Scaling afin d'utiliser les modèles de lancement au lieu des configurations de lancement. De plus, nous avons implément é les dernières actions d'automatisation et les derniers environnements d'exécution dans le contenu du runbook.	22 juin 2022

Change Manager : empêcher les utilisateurs de créer des requêtes auto-approuvables	Vous pouvez désormais configurer des modèles de modifications dans Change Manager pour prendre en charge les approbations automatiques. Cela signifie que les utilisateurs disposant des autorisations IAM nécessaires peuvent choisir de démarrer la demande de modification sans avoir besoin d'une approbation supplémentaire. Vous pouvez aussi empêcher des utilisateurs individuels, des groupes ou des rôles IAM d'envoyer des demandes d'approbation automatique, même si un modèle de modification les prend en charge. Cela est obtenu en utilisant une nouvelle clé de condition IAM, `ssm:AutoApprove` . Pour plus d'informations, consultez la rubrique Contrôle de l'accès aux flux de travail des runbooks d'approbation automatique	15 juin 2022

Conseils actualisés pour les rôles des tâches de la fenêtre de maintenance	Auparavant, la console Systems Manager vous permettait de choisir le rôle `AWSServiceRoleForAmazonSSM` lié au service IAM géré par AWS à utiliser comme rôle de maintenance pour vos tâches. L'utilisation de ce rôle et de la politique associée, `AmazonSSMServiceRolePolicy`, pour les tâches de la fenêtre de maintenance n'est plus recommandée. Vous devez plutôt créer une politique et un rôle personnalisés pour les tâches de fenêtre de maintenance. Pour plus d'informations, consultez Configuration de Maintenance Windows.	9 juin 2022
Prise en charge du transfert de port vers des hôtes distants pour Session Manager	Session Manager prend désormais en charge les sessions de transfert de port vers des hôtes distants. L'hôte distant ne doit pas nécessairement être géré par Systems Manager. Pour de plus amples informations, consultez la rubrique Starting a session (port forwarding to remote host) (Démarrage d'une session (réacheminement de port vers l'hôte distant)).	25 mai 2022

Contenu mis à jour : instructions d'installation manuelle de SSM Agent sur les instances Linux Amazon EC2	En réponse aux commentaires des clients, nous avons remanié les rubriques qui fournissent des instructions d'installation manuelle de SSM Agent sur les instances Amazon EC2. Ces rubriques fournissent désormais des commandes utilisant des fichiers disponibles dans le monde entier que vous pouvez copier et coller pour une installation rapide sur des instances EC2 dans n'importe quelle Région AWS. Ces rubriques fournissent également des informations pour vous aider à créer des commandes d'installation utilisant des fichiers disponibles dans votre propre région de travail. Cette dernière approche est recommandée lorsque vous installez l'agent sur plusieurs instances à l'aide d'un script ou d'un modèle. Pour plus d'informations, consultez les instructions fournies pour votre système d'exploitation Linux dans la rubrique Installation manuelle de SSM Agent sur les instances EC2 pour Linux.	9 mai 2022

Nouvelle rubrique : Amazon Machine Images (AMIs) avec SSM Agent préinstallé	En réponse aux commentaires des clients, nous disposons d'informations centralisées sur les AMIs gérées par AWS sur lesquelles SSM Agent est préinstallé. Cette rubrique fournit également des instructions sur la façon de vérifier qu'une instance Amazon EC2 a été créée à partir de ces AMIs a été correctement installée et est en cours d'exécution. Dans de rares cas où l'agent peut ne pas s'installer correctement ou ne pas démarrer, nous fournissons également des informations sur le démarrage ou l'installation manuelle de l'agent sur ces instances. Pour plus d'informations, consultez la rubrique Amazon Machine Images (AMIs) avec SSM Agent préinstallé.	8 mai 2022
Nouvelle section State Manager	Ajout d'une nouvelle section qui décrit les détails concernant le moment où State Manager exécute des associations. Pour plus d'informations, consultez la rubrique À propos de la planification des associations.	27 avril 2022

Patch Manager prend désormais en charge Rocky Linux	Vous pouvez désormais utiliser Patch Manager pour appliquer les correctifs les nœuds Rocky Linux. Bon nombre des règles qui s'appliquent à RHEL 8 pour les correctifs s'appliquent également à Rocky Linux. Rocky Linux 8 utilise le nouveau `AWS-DefaultRockyLinuxPatchBaseline` . Pour plus d'informations, consultez les rubriques suivantes : • Sélection des correctifs de sécurité • Installation des correctifs • Fonctionnement des règles de référentiel de correctifs sur RHEL, CentOS Stream et Rocky Linux.	14 avril 2022

Patch Manager prend désormais en charge CentOS Stream 8	Vous pouvez désormais utiliser Patch Manager pour appliquer les correctifs aux instances CentOS Stream 8 et Red Hat Enterprise Linux (RHEL) versions 4.4–4.5. Bon nombre des règles qui s'appliquent à RHEL 8 pour les correctifs s'appliquent également à CentOS Stream. CentOS Stream 8 utilise le nouveau `AWS-DefaultCentOSPatchBaseline` . Pour plus d'informations, consultez les rubriques suivantes : • Sélection des correctifs de sécurité • Installation des correctifs • Fonctionnement des règles de référentiel de correctifs sur RHEL et CentOS Stream	4 avril 2022

Créer un rôle de responsable pour Change Manager	Une nouvelle section détaille les exigences relatives à la création et à la mise en œuvre d'un Rôle de responsable pour Change Manager. Un rôle de responsable est une fonction du service (IAM) AWS Identity and Access Management qui permet à Change Manager d'exécuter en toute sécurité les flux de travail du Runbook spécifiés dans une demande de modification approuvée en votre nom. Le rôle accorde la `AssumeRole` confiance à AWS Systems Manager (AWS STS)Change Manager. Pour plus d'informations, consultez Configuration de rôles et autorisations pour Change Manager.	18 mars 2022
Approbation ou rejet de plusieurs demandes de modification Change Manager	Dans la console Systems Manager, vous pouvez désormais sélectionner plusieurs demandes de modification à approuver ou à rejeter en une seule opération . Pour plus d'informations, consultez Vérifier et approuver ou rejeter les demandes de modifications (console).	8 mars 2022

Prise en charge de Rocky Linux et des nœuds gérés Windows Server 2022	Systems Manager prend en charge Rocky Linux et les nœuds gérés 2022 Windows Server, y compris les dispositifs périphériques et les machines hybrides situés sur site ou logés sur d'autres fournisseurs de services de cloud. Pour utiliser Systems Manager avec ces systèmes d'exploitation, vous devez effectuer toutes les procédures de configuration requises par Systems Manager, y compris les procédures pour les environnements hybrides ou les dispositifs périphériques, le cas échéant. Pour plus d'informations, consultez Configuration de Systems Manager. Pour les ordinateurs Rocky Linux, vous devez également installer manuellement SSM Agent. Pour plus d'informations, consultez Installer manuellement SSM Agent sur les instances Rocky Linux. Pour les instances Windows Server 2022, Amazon Elastic Compute Cloud (Amazon EC2) SSM Agent est installé par défaut.	1er mars 2022

Permettez à Automation de s'adapter à vos besoins de simultanéité et de consulter les indicateurs d'utilisation de l'automatisation	Vous pouvez désormais autoriser Automation à ajuster automatiquement votre quota d'automatisation simultané e et à consulter les statistiq ues d'utilisation de l'automat isation publiées sur CloudWatc h. Pour plus d'informations sur la simultanéité adaptative, consultez Autoriser Automatio n à s'adapter à vos besoins de simultanéité. Pour plus d'informations sur la façon de consulter les statistiq ues d'utilisation de l'automat isation, consultez la section Surveillance des métriques d'automatisation à l'aide d'Amazon CloudWatch.	27 janvier 2022

Permettez à Automation de s'adapter à vos besoins de simultanéité et de consulter les indicateurs d'utilisation de l'automatisation	Vous pouvez désormais autoriser Automation à ajuster automatiquement votre quota d'automatisation simultané e et à consulter les statistiq ues d'utilisation de l'automat isation publiées sur CloudWatc h. Pour plus d'informations sur la simultanéité adaptative, consultez Autoriser Automatio n à s'adapter à vos besoins de simultanéité. Pour plus d'informations sur la façon de consulter les statistiq ues d'utilisation de l'automat isation, consultez la section Surveillance des métriques d'automatisation à l'aide d'Amazon CloudWatch.	27 janvier 2022
Documents Systems Manager organisés par catégories	Les documents Systems Manager appartenant à Amazon sont désormais organisés par type et par catégories pour vous aider à trouver les documents dont vous avez besoin.	13 janvier 2022

Créer et appeler des intégrations pour Automation	Vous pouvez désormais envoyer des messages à l'aide de webhooks pendant une automatisation en créant une intégration. Les intégrations peuvent être appelées lors d'une automatisation à l'aide de la nouvelle action `aws:invokeWebhook` dans votre runbook. Pour plus d'informations sur la création d'intégrations, consultez la rubrique Creating webhook integrations for Automation (Création d'intégrations webhook pour Automation). Pour en savoir plus sur l'action `aws:invokeWebhook` , consultez `aws:invokeWebhook` : appeler une intégration de webhook Automation.	13 janvier 2022

Fonctionnalités non disponibles dans les nouvelles versions Région AWS	Les fonctionnalités de Systems Manager suivantes ne sont actuellement pas disponibles dans la nouvelle région Asie-Pacifique (Jakarta) . • Application Manager • Change Calendar • Change Manager • Explorer • Fleet Manager • Incident Manager • Quick Setup	13 décembre 2021

Afficher les détails liés au coût des ressources d'une application	Application Managerest intégré AWS Billing and Cost Management via le widget Cost Explorer. Une fois que vous avez activé explorateur de coûts dans la console de facturation et gestion des coûts, le widget explorateur de coûts d'Application Manager affiche les données de coût d'une application ou d'un composant d'application spécifique non conteneurisé. Vous pouvez appliquer des filtres dans le widget pour afficher les données de coût selon différentes périodes, différentes granularités et différents types sous forme de graphique à barres ou linéaire. Pour plus d'informations, consultez Viewing overview information about an application.	7 décembre 2021
Gestion des processus à l'aide de Fleet Manager	Vous pouvez désormais utiliser Fleet Manager pour gérer les processus sur vos nœuds. Pour plus d'informations, consultez Utilisation des processus.	6 décembre 2021

Modification terminologique : les instances gérées sont désormais des nœuds gérés	Avec la prise en charge des appareils AWS IoT Greengrass principaux, l'expression « instance gérée » a été remplacée par « nœud géré » dans la majeure partie de la documentation de Systems Manager. La console Systems Manager, les appels d'API, les messages d'erreur et les documents SSM utilisent toujours le terme instance.	29 novembre 2021

Prise en charge des appareils de périphérie	Systems Manager prend en charge les configurations d'appareils de périphérie suivantes. • AWS IoT Greengrass : Systems Manager prend désormais en charge tous les appareils configurés AWS IoT Greengrass et exécutant le logiciel AWS IoT Greengrass Core. Pour intégrer vos appareils AWS IoT Greengrass principaux, vous devez créer un rôle de service AWS Identity and Access Management (IAM). Vous devez également utiliser la AWS IoT Greengrass console pour le déployer SSM Agent en tant que AWS IoT Greengrass composant sur vos appareils. Pour plus d'informations, consultez la section Configuration AWS Systems Manager des appareils Edge. • Appareils Edge dans un environnement hybride : Systems Manager prend également en charge les appareils AWS IoT principaux et les appareils non AWS IoT une fois que vous les avez configurés en tant que	29 novembre 2021

	machines sur site. Pour intégrer vos appareils, vous devez créer une fonction de service IAM, créer une activation de nœud géré pour un environnement hybride et installer manuellement l'SSM Agent sur vos appareils. Pour plus d'informations, voir Configuration AWS Systems Manager pour les environnements hybrides	
Connexion aux instances gérées à l'aide du Bureau à distance	Vous pouvez désormais utiliser Fleet Manager pour vous connecter aux instances Windows à l'aide du protocole RDP (Remote Desktop Protocol). Ces sessions de Bureau à distance optimisées par NICE DCV permettent des connexions sécurisées à vos instances à partir de votre navigateur. Pour plus d'informations, consultez Connexion à l'aide du protocole RDP.	23 novembre 2021

Spécification d'une durée de session maximale et détail des raisons des sessions	Vous pouvez désormais spécifier une durée de session maximale pour toutes les session Session Manager d'une Région AWS de votre Compte AWS. Lorsqu'une session atteint la durée spécifiée, elle est prend fin. Vous pouvez également ajouter une raison au démarrage d'une session. Pour plus d'informations, consultez Spécification d'une durée de session maximale.	16 novembre 2021

Patch Manager prend désormais en charge le système d'exploitation Raspberry Pi OS	Vous pouvez maintenant utiliser Patch Manager pour appliquer le correctif aux instances de Raspberry Pi OS. Patch Manager prend en charge les correctifs Raspberry Pi OS 9 (Stretch) et 10 (Buster). Le système d'exploitation Raspberry Pi OS étant basé sur Debian, de nombreuses règles de correctif identiques s'y appliquent comme à Debian Server. Pour plus d'informations, consultez les rubriques suivantes : • Sélection des correctifs de sécurité • Installation des correctifs • Fonctionnement des règles de référence de correctifs sur Debian Server et Raspberry Pi OS	16 novembre 2021
Accès au portail de la base de connaissances Red Hat	Utilisez Fleet Manager pour accéder au portail de la base de connaissances RHEL afin de trouver des solutions, des articles, de la documentation et des vidéos sur l'utilisation des produits Red Hat. Pour plus d'informations, consultez Accès au portail de la base de connaissances Red Hat.	3 novembre 2021

Modification en bloc de OpsItems	OpsCenter prend désormais en charge la modification en bloc de OpsItems. Vous pouvez sélectionner plusieurs OpsItems et modifiez l'un des champs suivants : Status (État), Priority (Priorité), Severity (Sévérité) et Category (Catégorie). Pour plus d'informations, consultez Modification d'OpsItems.	15 octobre 2021
Créez des paramètres d'entrée qui renseignent les ressources AWS	Vous pouvez désormais créer des paramètres d'entrée dans les runbooks Automatio n destinés à renseigner les ressources AWS dans la AWS Management Console. Pour plus d'informations, consultez la section Création de paramètres d'entrée qui renseignent les AWS ressources.	14 octobre 2021
Nouvelle option de limitation d'invocation pour les fenêtres de maintenance	Vous pouvez désormais choisir de bloquer tout nouvel appel de tâches lorsque l'heure limite spécifiée pour une fenêtre de maintenan ce est atteinte. Pour plus d'informations, consultez Attribution de tâches à une fenêtre de maintenance (console).	13 octobre 2021

Prise en charge de Patch Manager pour macOS 11.3.1 et 11.4 (Big Sur)	Les instances Amazon Elastic Compute Cloud (Amazon EC2) pour macOS 11.3.1 et 11.4 (Big Sur) peuvent désormais être corrigées à l'aide de Patch Manager. Cela s'ajoute à la prise en charge existante pour macOS 10.14.x (Mojave) et 10.15.x (Catalina). Pour plus d'informations sur l'utilisation de Patch Manager, consultez AWS Systems ManagerPatch Manager	1er octobre 2021

Application insights dans Application Manager	Application Managers'intègre à Amazon CloudWatch Application Insights. Application Insights identifie et paramètre des métriques, des journaux et des alarmes clés dans vos ressources d'application et votre pile technologique. Application Insights surveille en permanence les métriques et les journaux afin de détecter et de corréler les anomalies et les erreurs. Lorsque le système détecte des erreurs ou des anomalies, Application Insights génère CloudWatch des événements que vous pouvez utiliser pour configurer des notifications ou prendre des mesures. Vous pouvez activer et afficher Application Insights sous les onglets Overview (Présentation) et Monitoring (Surveillance) dans Application Manager. Pour plus d'informations sur Application Insights, consultez la section Qu'est-ce qu'Amazon CloudWatch Application Insights dans le guide de CloudWatch l'utilisateur Amazon.	21 septembre 2021

Importer des événements d'autres calendriers dans Change Calendar	Vous pouvez désormais importer les événements d'un calendrier tiers dans un calendrier dans Change Calendar. Auparavant, chaque événement devait être saisi manuellement dans un calendrier. Après avoir exporté un calendrier d'un fournisseur de calendrier tiers pris en charge vers un fichier iCalendar (`.ics`), importez-le dans Change Calendar. Ses événements sont inclus dans les règles de votre calendrier ouvert ou fermé dans Systems Manager. Les fournisseurs pris en charge comprenne nt iCloud Calendar, Google Calendar et Microsoft Outlook. Pour de plus amples informati ons, consultez Importation et gestion d'événements à partir de calendriers tiers.	8 septembre 2021

Nouvelles fonctions de balisage et de runbook dans Application Manager	Les améliorations apportées au balisage incluent la possibilité d'ajouter ou de supprimer des balises d'une ressource spécifique ou de toutes les ressources d'une application Application Manager. Les améliorations apportées au runbook incluent la possibilité d'afficher une liste filtrée de runbooks pour un type de ressource spécifique ou de lancer un runbook sur toutes les ressources du même type. Pour de plus amples informations, consultez Utilisation des balises dans Application Manager et Utilisation de runbooks dans Application Manager.	31 août 2021
Nouvel exemple : créez une demande de modification à l'aide du AWS CLI	Un exemple de création d'une demande de modification avec le AWS CLI a été ajouté au Change Manager chapitre. L'exemple utilise l'exemple de modèle de modification `AWS-HelloWorldChangeTemplate` et `AWS-HelloWorld runbook` : • Création de demandes de modifications (AWS CLI)	20 août 2021

Nouvelle section : utiliser les paramètres dans Amazon EKS	Une nouvelle section a été ajoutée au chapitre Parameter Store. Cette rubrique démontre comment utiliser vos paramètres dans des clusters Amazon EKS. Pour de plus amples informations, consultez Utiliser des paramètres Parameter Store dans Amazon Elastic Kubernetes Service.	19 août 2021
Hooks de cycle de vie Patch Manager mis à jour	Patch Manager fournit désormais un hook de cycle de vie (c-à-d la possibilité d'exécuter un document de commande Systems Manager) pour un point supplémentaire durant une opération d'application de correctifs Corriger maintenant. Si vous planifiez des redémarrages d'instance après l'exécution d'une opération Corriger maintenant, vous pouvez spécifier un hook de cycle de vie à exécuter une fois le redémarrage terminé. Pour de plus amples informations, consultez Utilisation de hooks de cycle de vie « Corriger maintenant » et À propos du Document SSM `AWS-RunPatchBaselineWithHooks`.	9 août 2021

Approbations automatiques maintenant prises en charge pour des demandes Change Manager	Vous pouvez désormais configurer des modèles de modifications dansChang e Manager pour prendre en charge les approbations automatiques. Cela signifie que les utilisateurs disposant des autorisations IAM nécessaires peuvent choisir de démarrer la demande de modification sans avoir besoin d'une approbation supplémentaire. Les utilisate urs qui ont accès aux modèles d'approbation automatique peuvent toujours choisir de spécifier des approbateurs s'ils le souhaitent. Pour vous aider à contrôler vos processus Change Manager, des approbations restent exigées pour toutes les demandes pendant les périodes de gel des modifications. Pour plus d'informations, consultez les rubriques suivantes : • Création de modèles de modification • Création de demandes de modifications • Essayez le modèle de AWS gestion des `Hello World` modifications	30 juillet 2021

Informations opérationnelles OpsCenter	OpsCenter analyse automatiquement des OpsItems dans votre compte et génère des informations. Une information comprend des informations utiles pour comprendre combien votre compte contient d'OpsItems en double et les sources qui les créent. Les informations fournissent aussi des bonnes pratiques recommandées et des runbooks Automation pour vous aider à résoudre les OpsItems en double. Pour de plus amples informations, consultez Utilisation des informations opérationnelles.	13 juillet 2021
Afficher les instances arrêtées dans Fleet Manager	Vous pouvez désormais afficher les instances `running` et les instances `stopped` à partir de la console Fleet Manager. Pour plus d'informations, consultez AWS Systems ManagerFleet Manager.	12 Juillet 2021
Nouvelle rubrique : création de runbooks Automation	Une nouvelle rubrique : Création de runbooks Automation, fournit des conseils et des exemples narratifs sur la création de contenu pour des runbooks Automation personnalisés.	8 juillet 2021

AWS CloudFormation création de piles et de modèles dans Application Manager	Application Managervous aide à provisionner et à gérer les ressources de vos applications en les intégrant à CloudFormation. Vous pouvez créer, modifier et supprimer des AWS CloudFormation modèles et des piles dansApplication Manager. Application Managerinclut également une bibliothèque de modèles dans laquelle vous pouvez cloner, créer et stocker des modèles. Application Manageret CloudFormation affichent les mêmes informations sur l'état actuel d'une pile. Les modèles et les mises à jour des modèles sont stockés dans Systems Manager jusqu'à ce que vous approvisionniez la pile, date à laquelle les modifications sont également affichées CloudFormation. Pour plus d'informations, consultez la section Travailler avec des AWS CloudFormation piles dans Application Manager.	8 juillet 2021

Nouvelle rubrique : rotation automatique des clés privées pour SSM Agent sur des instances hybrides	Une nouvelle rubrique : Configuration de la rotation automatique des clés privées, fournit des instructions sur le renforcement de votre posture de sécurité en configurant SSM Agent de sorte à assurer la rotation automatique des clés privées de l'environ nement hybride.	15 juin 2021
Session Managerplugin pour la AWS CLI version 1.2.205.0	Une nouvelle version du Session Manager plugin pour le AWS CLI a été publiée. Pour de plus amples informati ons, consultez Dernière version de plugin Session Manager et historique des versions.	10 juin 2021

Nouveau rôle lié au service IAM	Lorsque vous activez les informations opérationnelles OpsCenter, Systems Manager crée un nouveau rôle lié au service AWS Identity and Access Management (IAM) appelé `AWSSSMOpsInsightsServiceRolePolicy` . Pour plus d'informations sur ce rôle, consultez la section Utilisation des rôles pour créer des informations opérationnelles OpsItems dans Systems Manager OpsCenter : `AWSSSMOpsInsightsServiceRolePolicy` .	9 juin 2021
Nouveau contenu de résolution des problèmes Patch Manager pour Linux	Une nouvelle rubrique : Erreurs lors de l'exécution de `AWS-RunPatchBaseline` sous Linux, fournit des descriptions et des solutions pour plusieurs problèmes pouvant être rencontrés lors de l'application de correctifs aux instances gérées avec les systèmes d'exploitation Linux.	8 juin 2021

Prise en charge améliorée des tâches de fenêtre de maintenance ne nécessitant pas de cibles spécifiées (console)	Vous pouvez désormais créer des tâches de fenêtre de maintenance dans la console sans spécifier de cible dans la tâche si cela n'est pas nécessaire. Auparavant, cette option n'était disponible que lors de l'utilisation de l'API AWS CLI or. Cette option s'applique à l'automatisation et aux types de AWS Step Functions tâches. AWS Lambda Par exemple, si vous créez une tâche Automation et que les ressources à mettre à jour sont spécifiées dans les paramètres du document Automation, vous n'avez plus besoin de spécifier une cible dans la tâche elle-même. Pour plus d'informations, consultez les rubriques Enregistrement de tâches de fenêtre de maintenance sans cibles, Attribuer des tâches à une fenêtre de maintenance (console) et Schedule automations with maintenance windows (Planifier des automatisations avec les fenêtres de maintenance).	28 mai 2021

Référence du runbook Automation délocalisée	La référence de runbook Automation a été déplacée vers un nouvel emplacement. Pour de plus amples informations, consultez la Référence des runbooks Automation de Systems Manager.	10 mai 2021
AWS Systems Manager Incident Manager lancement	Incident Manager est une console de gestion des incidents conçue pour aider les utilisateurs à atténuer les incidents affectant leurs applications AWS hébergées et à s'en remettre. Pour plus d'informations, consultez le AWS Systems Manager Incident Manager Guide de l'utilisateur .	10 mai 2021
State Manager prend en charge Change Calendar	Vous pouvez désormais spécifier des noms ou des Amazon Resource Names (ARN) Change Calendar lorsque vous créez ou mettez à jour une association State Manager. State Manager applique des associations uniquement lorsque le calendrier des modifications est ouvert, pas quand il est fermé. Pour de plus amples informations, consultez Création d'associations et Modification et création d'une version d'une association.	6 mai 2021

Clôner les documents Systems Manager	La console Systems Manager Documents vous permet désormais de copier le contenu d'un document existant vers un nouveau document, que vous pouvez modifier. Pour en savoir plus, consultez Clonage d'un document SSM.	4 mai 2021
Intégrer Security Hub à Explorer et OpsCenter	Vous pouvez désormais intégrer Explorer et OpsCenter avec AWS Security Hub. Security Hub fournit une vue complète de l'état de votre sécurité AWS et vous aide à vérifier que votre environnement est conforme aux normes et aux meilleure s pratiques du secteur de la sécurité. Après l'intégra tion à Explorer, vous pouvez consulter les résultats de sécurité dans le widget Security Hub du tableau de bord Explorer. Après l'intégra tion à OpsCenter, vous pouvez créer des OpsItems pour des résultats Security Hub. Pour plus d'informations, consultez les sections Réception des résultats de AWS Security Hub l'entrée Explorer et Réception des conclusions de AWS Security Hub l'entrée OpsCenter.	27 avril 2021

Nouvelle rubrique : conventions de document	Une nouvelle rubrique aide les utilisateurs à se familiariser avec les conventions typographiques courantes utilisées dans le Guide de l'utilisateur AWS Systems Manager . Pour de plus amples informations, consultez Configuration de rédaction.	21 avril 2021
Rubrique mise à jour : à propos des opérations d'application de correctifs publiées par Microsoft sur Windows Server	La rubrique À propos de la correction d'applications publiées par Microsoft sur Windows Server précise désormais clairement que, pour que Patch Manager puisse corriger les applications publiées par Microsoft sur vos instances gérées par Windows Server, l'option de mise à jour Windows Give me updates for other Microsoft products when I update Windows (Me proposer des mises à jour pour d'autres produits Microsoft lorsque je mets à jour Windows) doit être autorisée sur l'instance.	12 avril 2021

Réorganisation de la référence des runbooks Automation	Pour vous aider à trouver les runbooks qu'il vous faut et à naviguer plus efficacement dans la référence, nous avons réorganisé le contenu de la référence des runbooks Automation par Service AWS. Pour afficher ces modifications, consultez Référence des runbooks Systems Manager Automation.	12 avril 2021

Patch Manager : générer des rapports de conformité des correctifs .csv	Patch Manager prend désormais en charge la génération de rapports de conformité des correctifs pour vos instances et l'enregistrement du rapport dans un compartiment S3 de votre choix, au format .csv. Ensuite, à l'aide d'un outil tel qu'Amazon QuickSight, vous pouvez analyser les données du rapport de conformité des correctifs. Vous pouvez générer un rapport de conformité des correctifs pour une instance unique ou pour toutes les instances de votre Compte AWS. Vous pouvez générer un rapport à la demande ponctuel ou configurer un calendrier pour la création automatique de rapports. Vous pouvez également spécifier une rubrique Amazon Simple Notification Service pour fournir des notifications lorsqu'un rapport est généré. Pour de plus amples informations, consultez Génération de rapports de conformité des correctifs CSV.	9 avril 2021

Supprimer des étiquettes de paramètre Parameter Store	Vous pouvez désormais supprimer des étiquettes de paramètres Parameter Store via la console Systems Manager ou la AWS CLI. Pour de plus amples informations, consultez Utilisation des étiquettes de paramètres.	6 avril 2021
Planifier les redémarrages de l'instance lors de l'utilisation de la fonction Corriger maintenant	Patch Manager prend désormais en charge la planification d'un délai pour le redémarrage de vos instances après l'installation des correctifs à l'aide de la fonction Corriger maintenant. Cette fonction s'ajoute aux options existantes qui permettent de ne redémarrer les instances que si cela est nécessaire pour terminer l'installation d'un correctif, ou d'ignorer le redémarrage après l'opération d'application de correctifs. Pour de plus amples informations, consultez Application de correctifs sur les instances à la demande.	1 avril 2021

Nouvelle rubrique : découverte de paramètres publics	Parameter Storeles paramètres publics peuvent désormais être trouvés à l'aide de la console AWS CLI ou de Systems Manager. Pour de plus amples informations, consultez Recherche de paramètres publics.	1 avril 2021
Mises à jour de l'opération Corriger maintenant : stocker des journaux dans S3 et exécuter des hooks de cycle de vie	Lorsque vous exécutez l'opération Patch now (Corriger maintenant) de Patch Manager, vous pouvez choisir un compartiment S3 pour le stockage automatique des journaux d'application de correctifs. Vous pouvez en outre choisir d'exécuter des documents de commande Systems Manager (documents SSM) en tant que hooks de cycle de vie en trois points de l'opération : Avant l'installation, Après l'installation et À la sortie. Pour obtenir des informations, consultez Application de correctifs sur les instances à la demande.	31 mars 2021

Systems Manager signale désormais les modifications apportées à ses politiques AWS gérées	À compter du 24 mars 2021, les modifications apportées aux politiques gérées sont signalées dans la rubrique « Systems ManagerMises à jour des politiques AWS gérées ». La première modification répertoriée est l'ajout de la prise en charge de la Explorer capacité de créer des rapports OpsData et OpsItems à partir de plusieurs comptes et régions.	24 mars 2021

Explorerautorise automatiquement toutes les OpsData sources à synchroniser les données des ressources en fonction des comptes dans AWS Organizations	Lorsque vous créez une synchronisation des données de ressources, si vous choisissez l'une des AWS Organizations options, Systems Manager autorise automatiquement toutes les OpsData sources dans la zone sélectionnée Régions AWS pour tous Comptes AWS dans votre organisation (ou dans les unités organisationnelles sélectionnées). Cela signifie, par exemple, que même si vous n'y êtes pas autorisé Explorer Région AWS, si vous sélectionnez une AWS Organizations option pour la synchronisation des données de vos ressources, Systems Manager collecte automatiquement les données OpsData provenant de cette région. Pour de plus amples informations, consultez À propos des synchronisations de données de ressources sur plusieurs comptes et plusieurs régions.	24 mars 2021

Systems Manager Automation fournit une nouvelle variable système pour vos runbooks	Avec la nouvelle variable `global:AWS_PARTITION` système, vous pouvez spécifier la AWS partition dans laquelle se trouve une ressource lors de la création de vos runbooks. Pour de plus amples informations sur les variables système, consultez Variables système Automation.	18 mars 2021
Autoriser plusieurs niveaux d'approbation pour les demandes de modifications Change Manager	Lorsque vous créez un modèle de modification Change Manager, vous pouvez désormais exiger que plusieurs niveaux d'approbateurs octroient une autorisation pour l'exécution d'une demande de modification. Par exemple, vous pouvez d'abord demander à des vérificateurs techniques d'approuver une demande de modification créée à partir d'un modèle de modification, puis exiger un second niveau d'approbation de la part d'un ou plusieurs responsables. Pour plus d'informations, consultez Création de modèles de modification.	4 mars 2021

Patch Manager prend désormais en charge Oracle Linux 8.x	Vous pouvez désormais utiliser Patch Manager pour appliquer le correctif aux instances Oracle Linux 8.x, via la version 8.3. Pour plus d'informations, consultez les rubriques suivantes : • Sélection des correctifs de sécurité • Installation des correctifs • Fonctionnement des règles de référentiel de correctifs sur Oracle Linux	1er mars 2021
OpsCenter affiche d'autres OpsItems pour une ressource sélectionnée	Pour vous aider à étudier les problèmes et à fournir le contexte d'un problème, vous pouvez consulter la liste OpsItems d'une AWS ressource spécifique. La liste affiche le statut, la sévérité et le titre de chaque OpsItem. La liste inclut également des liens profonds vers chaque OpsItem. Pour de plus amples informations, consultez Affichage d'autres OpsItems pour une ressource spécifique.	1er mars 2021

Définir les préférences d'application de correctifs au moment de l'exécution	Vous pouvez définir les préférences d'application de correctifs au moment de l'exécution en utilisant la fonction de remplacement de référentiel. Pour plus d'informations, consultez la section Utilisation du BaselineOverride paramètre.	25 février 2021
Nouveau type de document Systems Manager	AWS CloudFormation les modèles peuvent désormais être stockés sous forme de documents Systems Manager. Le stockage CloudFormation de modèles sous forme de documents Systems Manager vous permet de bénéficier des fonctionnalités des documents de Systems Manager telles que le contrôle des versions, la comparaison du contenu des versions et le partage avec des comptes. Pour plus d'informations, consultez Documents AWS Systems Manager.	9 février 2021

Corriger des instances en utilisant des hooks facultatifs	Le nouveau document SSM `AWS-RunPatchBaselineWithHooks` fournit des hooks pour exécuter des documents SSM en trois points du cycle d'application de correctifs sur les instances. Pour obtenir des informations sur `AWS-RunPatchBaselineWithHooks`, consultez À propos du document SSM `AWS-RunPatchBaselineWithHooks`. Pour obtenir un exemple de démonstration d'une opération d'application de correctifs utilisant les trois hooks, consultez Démonstration : mettre à jour les dépendances de l'application, corriger une instance et effectuer une surveillance de l'état spécifique à l'application.	2 février 2021

Nouvelle rubrique : validation des serveurs et des machines virtuelles sur site à l'aide d'une empreinte matérielle	SSM Agent vérifie l'identification des serveurs et des machines virtuelles sur site, que vous enregistrez auprès du service à l'aide d'une empreinte digitale calculée. L'empreinte digitale est une chaîne opaque, stockée dans le Coffre-fort que l'agent transmet à certaines API Systems Manager. Pour obtenir des informations sur l'empreinte matérielle et les instructions de configuration d'un seuil de similitude pour faciliter la vérification de la machine, consultez Validation des serveurs et des machines virtuelles sur site à l'aide d'une empreinte matérielle.	25 janvier 2021
Nouvelle rubrique : référence technique SSM Agent	La rubrique « Référence SSM Agent technique » rassemble des informations qui vous aideront à implémenter AWS Systems Manager SSM Agent et à comprendre le fonctionnement de l'agent. Cette rubrique inclut une toute nouvelle section contenant des mises à SSM Agent jour régulières par Régions AWS.	21 janvier 2021

SSM Agent sur Windows Server 2008	Depuis le 14 janvier 2020, Windows Server 2008 n'est plus pris en charge pour les mises à jour de fonctions ou de sécurité de Microsoft. Les AMIs Windows Server 2008 incluent effectivement SSM Agent, mais l'agent n'est plus mis à jour pour ce système d'exploitation.	5 janvier 2021

Support amélioré pour les tâches de la fenêtre de maintenance qui ne nécessitent pas de cibles spécifiées (AWS CLI et d'API uniquement)	Vous pouvez désormais créer des tâches pendant la fenêtre de maintenance sans avoir à spécifier de cible dans la tâche si aucune cible n'est requise (AWS CLI et uniquement sur l'API). Cela s'applique à l'automatisation AWS Lambda et aux types de AWS Step Functions tâches. Par exemple, si vous créez une tâche Automation et que les ressources à mettre à jour sont spécifiées dans les paramètres du runbook Automation, vous n'avez plus besoin de spécifier une cible dans la tâche elle-même. Pour plus d'informations, consultez les rubriques Enregistrement de tâches de fenêtre de maintenance sans cibles et Schedule automations with maintenance windows (Planifier des automatisations avec les fenêtres de maintenance).	23 décembre 2020

Nouvelles fonctions Automation	Une nouvelle propriété partagée a été ajoutée aux runbooks Systems Manager Automation. La propriété `onCancel` vous permet de spécifier quelle étape l'automatisation doit atteindre dans le cas où un utilisateur annule l'automatisation. Pour de plus amples informations, consultez Propriétés partagées par toutes les actions.	21 décembre 2020
Nouvelle rubrique : utilisation d'associations avec IAM	Une nouvelle rubrique a été ajoutée au chapitre Systems Manager State Manager qui décrit les bonnes pratiques pour créer des associations avec IAM. Pour plus d'informations, consultez Utilisation d'associations à l'aide d'IAM.	18 décembre 2020
State Manager prend désormais en charge plusieurs régions et plusieurs comptes	Les associations peuvent désormais être créées ou mises à jour avec plusieurs régions ou plusieurs comptes. Pour plus d'informations, consultez Création d'associations.	15 décembre 2020

Nouvelle fonctionnalité : Fleet Manager	Fleet Manager, une fonctionnalité de AWS Systems Manager, est une expérience d'interface utilisateur (UI) unifiée qui vous aide à gérer à distance votre parc de serveurs fonctionnant sur AWS site ou sur site. Avec Fleet Manager, vous pouvez consulter l'état et le statut de performance de votre flotte de serveurs à partir d'une console unique. Vous pouvez également collecter des données provenant d'instances individuelles pour effectuer des tâches courantes de résolution des problèmes et de gestion à partir de la console. Pour plus d'informations, consultez AWS Systems Manager.Fleet Manager	15 décembre 2020

Nouvelle fonctionnalité : Change Manager	Amazon Web Services a publié Change Manager, un cadre de gestion des modifications d'entreprise servant à demander, approuver, mettre en œuvre et générer des rapports sur les modifications opérationnelles apportées à la configuration et à l'infrastructure de votre application. À partir d'un seul compte d'administrateur délégué, si vous en avez un AWS Organizations, vous pouvez gérer les modifications sur plusieurs Comptes AWS comptes Régions AWS. En variante, en utilisant un compte local, vous pouvez gérer les modifications d'un Compte AWS unique. Change ManagerÀ utiliser pour gérer les modifications apportées aux AWS ressources et aux ressources locales. Pour plus d'informations, consultez AWS Systems Manager.Change Manager	15 décembre 2020

Nouvelle fonctionnalité : Application Manager	Application Managervous aide à étudier et à résoudre les problèmes liés à vos AWS ressources dans le contexte de vos applications. Application Managerregroupe les informations d'exploitation issues de plusieurs fonctionnalités Services AWS et de Systems Manager en une seule AWS Management Console. Pour plus d'informations, consultez AWS Systems Manager.Application Manager	15 décembre 2020

AWS Systems Manager prend en charge les instances Amazon EC2 pour macOS	Conjointement avec la prise en charge d'instances macOS par Amazon Elastic Compute Cloud (Amazon EC2), Systems Manager prend désormais en charge de nombreuses opération s sur des instances EC2 pour macOS. Les versions prises en charge comprenne nt macOS 10.14.x (Mojave) et 10.15.x (Catalina). Pour plus d'informations, consultez les rubriques suivantes. • Pour plus d'informations sur l'installation de l'SSM Agent sur des instances EC2 pour macOS, consultez Installat ion et configuration de l'SSM Agent sur des instances EC2 pour macOS. • Pour plus d'informations sur la correction des instances EC2 pour macOS, consultez Installation des correctifs et Création d'un référentiel de correctifs personnalisé (macOS). • Pour obtenir des informati ons générales sur la prise en charge des instances EC2 pourmacOS, consultez les instances Mac Amazon EC2 dans le guide de l'utilisateur Amazon EC2.	30 novembre 2020

Pseudo paramètres de fenêtre de maintenance : un nouveau type de ressource pris en charge pour {{TARGET_ID}} et {{RESOURCE_ID}}	Un type de ressource supplémentaire est désormais disponible pour une utilisation avec les pseudo-paramètres {{TARGET_ID}} et {{RESOURCE_ID}} . Vous pouvez désormais utiliser le type de ressource AWS::RDS::DBCluster avec ces deux pseudo-paramètres. Pour plus d'informations sur les pseudo-paramètres de la fenêtre de maintenance, consultez la section Utilisation de pseudo-paramètres lors de l'enregistrement des tâches de la fenêtre de maintenance.	27 novembre 2020
Session Managerplugin pour la AWS CLI version 1.2.30.0	Une nouvelle version du Session Manager plugin pour le AWS CLI a été publiée. Pour de plus amples informations, consultez Dernière version de plugin Session Manager et historique des versions.	24 novembre 2020
Nouvelle rubrique : comparaison des versions de documents SSM	Vous pouvez désormais comparer les différences de contenu entre les versions de documents SSM dans la console Systems Manager Documents. Pour de plus amples informations, consultez Comparaison des versions de documents SSM.	24 novembre 2020

Systems Manager prend désormais en charge les politiques de point de terminaison VPC	Vous pouvez désormais créer des politiques pour les points de terminaison d'interface VPC pour Systems Manager. Pour de plus amples informations, consultez Créer une politique de point de terminaison de VPC d'interface.	18 novembre 2020
Nouvelle rubrique : spécification d'une valeur d'expiration d'une session inactive	Vous pouvez maintenant spécifier le temps nécessaire à un utilisateur pour passer à l'état inactif avant qu'une session se termine avec Session Manager. Pour de plus amples informations, consultez Spécification d'une valeur d'expiration d'une session inactive.	18 novembre 2020
Nouvelle fonction de journalisation Session Manager	Vous pouvez désormais envoyer un flux continu de journaux de données de session au format JSON à Amazon Logs. CloudWatch Pour plus d'informations, consultez la section Données de session de streaming à l'aide d'Amazon CloudWatch Logs.	18 novembre 2020

Nouvelle rubrique : vérification de la signature de l'SSM Agent	Vous pouvez maintenant vérifier la signature cryptographique du package d'installation pour l'SSM Agent sur les instances Linux. Pour de plus amples informations, consultez Schémas et fonctions des documents SSM.	17 novembre 2020
Nouvelle rubrique : comprendre les statuts d'automatisation	Une nouvelle rubrique a été ajoutée au chapitre Systems Manager Automation qui décrit les statuts des actions et des automatisations. Pour de plus amples informations, consultez Comprendre les statuts d'automatisation.	17 novembre 2020
Nouveaux types de source pour le plugin `aws:downloadContent`	Git et HTTP sont maintenant pris en charge en tant que types de sources pour le plugin `aws:downloadContent`. Pour plus d'informations, consultez `aws:downloadContent`.	17 novembre 2020
Nouvelle fonction de schéma de document Systems Manager (document SSM)	Dans les documents SSM avec la version de schéma 2.2 ou ultérieure, le paramètre `precondition` prend désormais en charge le référencement des paramètres d'entrée de votre document. Pour de plus amples informations, consultez Schémas et fonctions des documents SSM.	17 novembre 2020

Nouvelle source de données dans Explorer : AWS Config	Exploreraffiche désormais des informations sur la AWS Config conformité, notamment un résumé général des AWS Config règles conformes et non conformes, le nombre de ressources conformes et non conformes, ainsi que des informations spécifiques sur chacune d'entre elles (lorsque vous recherchez une règle ou une ressource non conforme). Pour de plus amples informations, consultez Modification des sources de données de Systems Manager Explorer.	11 novembre 2020
Nouvelle rubrique : exécution de groupes Auto Scaling avec des associations	Une nouvelle section a été ajoutée à State Manager, qui décrit les bonnes pratiques de création d'associations pour exécuter des groupes Auto Scaling. Pour de plus amples informations, consultez Exécution de groupes Auto Scaling avec des associations.	10 novembre 2020
Quick Setup prend désormais en charge le ciblage d'un groupe de ressources	Quick Setup prend désormais en charge le choix d'un groupe de ressources comme cible pour le type de configuration local. Pour de plus amples informations, consultez Choisir des cibles pour Quick Setup.	5 novembre 2020

Patch Manager ajoute la prise en charge de Debian Server 10 LTS, Oracle Linux 7.9 LTS et Ubuntu Server 20.10 STR	Vous pouvez désormais utiliser Patch Manager pour appliquer le correctif aux instances Debian Server 10 LTS, Oracle Linux 7.9 LTS et Ubuntu Server 20.10 STR. Pour plus d'informations, consultez les rubriques suivantes : • Conditions préalables requises Patch Manager • Sélection des correctifs de sécurité • Installation des correctifs • Fonctionnement des règles de référentiel de correctifs sur Debian Server • Fonctionnement des règles de référentiel de correctifs sur Oracle Linux • Fonctionnement des règles de référentiel de correctifs sur Ubuntu Server	4 novembre 2020

Nouveau EventBridge support pour AWS Systems ManagerChange Calendar	Amazon fournit EventBridge désormais une assistance pour les Change Calendar événements dans les règles relatives aux événements. Lorsque l'état d'un calendrie r change, vous EventBridge pouvez lancer l'action cible que vous avez définie comme EventBridge règle. Pour plus d'informations sur l'utilisation des événements Systems Manager EventBridge et sur leur utilisation, consultez les rubriques suivantes. • Configuration EventBridge pour les événements de Systems Manager • Référence : modèles et types d' EventBridge événements Amazon pour Systems Manager	4 novembre 2020

Configurer CloudWatch pour créer à OpsItems partir d'alarmes	Vous pouvez configurer Amazon CloudWatch pour créer automatiquement un OpsItem dans Systems Manager OpsCenter lorsqu'une alarme entre dans l'`ALARM`état. Cela vous permet de diagnostiquer et de résoudre rapidement les problèmes liés aux AWS ressources à partir d'une console unique. Pour plus d'informations, voir Configuration CloudWatch pour créer à OpsItems partir d'alarmes.	4 novembre 2020
Prise en charge d'Ubuntu Server 20.10	AWS Systems Manager prend désormais en charge la version à court terme Ubuntu Server 20.10 (STR). Pour plus d'informations, consultez les rubriques suivantes : • Systèmes d'exploitation pris en charge • Installer SSM Agent pour un environnement hybride (Linux) • Installation manuelle de l'SSM Agent sur les instances Ubuntu Server • Vérification du statut de l'SSM Agent et démarrage de l'agent	22 octobre 2020

Nouvelle rubrique : autoriser les profils de shell configurables	Vous pouvez désormais autoriser les profils de shell configurables avec Session Manager. En autorisant les profils de shell configurables, vous pouvez personnaliser des préférences de session, telles que des préférences de shell, des variables d'environnement, des répertoires de travail et l'exécution de plusieurs commandes au démarrage d'une session. Pour de plus amples informations, consultez Autoriser les profils de shell configurables.	21 octobre 2020
Les résultats de conformité des correctifs indiquent désormais quels CVE sont résolus, et par quels correctifs	Pour la plupart des systèmes Linux pris en charge, lorsque vous affichez les résultats de conformité des correctifs pour vos instances gérées, vous pouvez consulter des détails indiquant les problèmes de bulletin CVE (Common Vulnerabilities and Exposure) qui sont résolus, et par quels correctifs disponibles. Ces informations peuvent vous aider à déterminer à quel point il est urgent d'installer un correctif manquant ou défaillant. Pour plus d'informations, consultez Affichage des résultats de conformité des correctifs.	20 octobre 2020

Prise en charge étendue des métadonnées de correctifs Linux	Vous pouvez désormais afficher de nombreux détails sur les correctifs Linux disponibles dans Patch Manager. Vous pouvez choisir d'afficher des 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'opération d'API DescribeAvailablePatches a été mise à jour pour prendre en charge les systèmes d'exploitation Linux et le filtrage en fonction de ces nouveaux types de métadonnées de correctifs disponibles. Pour plus d'informations, consultez les rubriques suivantes : • Affichage des correctifs disponibles • DescribeAvailablePatches et Patch dans la Référence d'API AWS Systems Manager • describe-available-patchesdans la AWS Systems Manager section de la référence de AWS CLI commande	16 octobre 2020

Session Managerplugin pour la AWS CLI version 1.2.7.0	Une nouvelle version du Session Manager plugin pour le AWS CLI a été publiée. Pour de plus amples informations, consultez Dernière version de plugin Session Manager et historique des versions.	15 octobre 2020
Nouvelle rubrique : schéma de document Session	La nouvelle rubrique Schéma de document Session décrit les éléments de schéma d'un document Session. Ces informations peuvent vous aider à créer des documents Session personnalisés où vous spécifiez des préférences pour les types de sessions que vous utilisez avec Session Manager.	15 octobre 2020
Nouvelle rubrique : recherche en texte libre pour documents SSM	La zone de recherche de la page Systems Manager Documents prend désormais en charge la recherche en texte libre. La recherche en texte libre compare le ou les termes de recherche saisis au nom de document dans chaque document SSM. Pour de plus amples informations, consultez Utilisation de la recherche en texte libre.	15 octobre 2020

Nouvelle rubrique : résolution des problèmes liés à la disponibilité d'instances gérées Amazon EC2	La nouvelle rubrique Résolution des problèmes liés à la disponibilité d'instances gérées Amazon EC2 vous aide à déterminer pourquoi une instance Amazon EC2 dont vous avez confirmé qu'elle était en cours d'exécution n'est pas disponible dans les listes d'instances gérées disponibles dans Systems Manager.	6 octobre 2020

Réorganisation du chapitre sur Parameter Store	Pour vous aider à trouver les informations nécessaires avec plus d'efficacité, nous avons réorganisé le contenu du chapitre Parameter Store du Guide de l'utilisateur AWS Systems Manager. La plupart des contenus sont désormais organisés dans les sections Configuration de Parameter Store et Utilisation de Parameter Store. La rubrique AWS Systems ManagerParameter Store a en outre été élargie de sorte à 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 ?	1er octobre 2020

Nouvelles rubriques relatives à la conformité des correctifs	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 mettre vos instances en conformité. • Identification des instances non conformes • Correction des instances non conformes • Affichage des résultats de conformité des correctifs	24 septembre 2020
SSM Agent version 3.0	Systems Manager a lancé une nouvelle version de SSM Agent.	21 septembre 2020

Rubriques nouvelles et mises à jour : Amazon EventBridge remplace CloudWatch Events pour la gestion des événements	CloudWatch Les événements et la même API EventBridge constituent le même service sous-jacent et la même API, mais ils EventBridge offrent davantage de fonctionnalités et constituent désormais le moyen préféré de gérer vos événements dans AWS. (Les modifications que vous apportez dans l'une CloudWatch ou l'autre console EventBridge sont reflétées dans chaque console.) Les références aux CloudWatch événements et aux procédures existantes dans le guide de AWS Systems Manager l'utilisateur ont été mises à jour pour refléter le EventBridge support. En outre, les nouvelles rubriques suivantes ont été ajoutées. • Surveiller les événements Systems Manager • Configuration EventBridge pour les événements de Systems Manager • Exemples de types cibles Systems Manager • Référence : modèles et types d' EventBridge événements Amazon pour Systems Manager	18 septembre 2020

Intégration AWS Security Hub et Patch Manager	Vous pouvez désormais intégrer Patch Manager à AWS Security Hub. Security Hub fournit une vue complète de l'état de votre sécurité AWS et vous aide à vérifier que votre environnement est conforme aux normes et aux meilleures pratiques du secteur de la sécurité. Lorsqu'il est intégré à Patch Manager, Security Hub surveille le statut de correction de vos flottes du point de vue de la sécurité. Pour plus d'informations, consultez la section Intégration Patch Manager avec AWS Security Hub.	17 septembre 2020

Pseudo paramètres de fenêtre de maintenance : des nouveaux types de ressources pris en charge pour `{{TARGET_ID}}` et `{{RESOURCE_ID}}`	Lorsque vous enregistrez une tâche de fenêtre de maintenance, 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 en utilisant la syntaxe du pseudo-paramètre, comme `{{TARGET_ID}}` et `{{RESOURCE_ID}}` . Une fois que la tâche de la 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` avec ces deux pseudo-paramètres. Pour plus d'informations sur les pseudo-paramètres de la fenêtre de maintenance, consultez la section Utilisation de pseudo-paramètres lors de l'enregis	14 septembre 2020

	trement des tâches de la fenêtre de maintenance.	
Corriger des instances à la demande avec la nouvelle option « Corriger maintenant »	Vous pouvez désormais utiliser la console Systems Manager pour corriger les instances ou rechercher des correctifs manquants à tout moment. Pour cela, inutile de créer ou de modifier un calendrier, ou spécifier des options de configuration d'application de correctifs complètes pour répondre à un besoin immédiat d'application de correctifs. Vous n'avez qu'à spécifier si vous voulez analyser ou installer des correctifs, et identifier les instances cibles pour l'opération. Patch Manager applique automatiquement le référentiel de correctifs par défaut actuel pour vos types d'instance et applique les options de bonnes pratiques pour le nombre d'instances qui sont corrigées en même temps et le nombre d'erreurs autorisées avant que l'opération échoue. Pour obtenir des informations, consultez Application de correctifs sur les instances à la demande.	9 septembre 2020

Nouvelle rubrique : vérification du statut de l'SSM Agent et démarrage de l'agent	La nouvelle rubrique Vérification du statut de l'SSM Agent et démarrage de l'agent fournit des commandes pour vérifier si l'SSM Agent est en cours d'exécution sur chaque système d'exploitation le prenant en charge. Elle fournit aussi les commandes nécessaires pour démarrer l'agent s'il n'est pas en cours d'exécution.	7 septembre 2020
Patch Manager prend désormais en charge Ubuntu Server 20.04 LTS	Vous pouvez désormais utiliser Patch Manager pour appliquer le correctif aux instances Ubuntu Server 20.04 LTS. Pour plus d'informations, consultez les rubriques suivantes : • Sélection des correctifs de sécurité • Installation des correctifs • Fonctionnement des règles de référentiel de correctifs sur Ubuntu Server	31 août 2020

Nouvelle rubrique pour Cas d'utilisation et bonnes pratiques	Nous avons ajouté une nouvelle rubrique pour aider les utilisateurs à appréhender rapidement les différences entre une Maintenance Windows et State Manager. Pour de plus amples informations, consultez Choisir entre State Manager et les Maintenance Windows.	28 août 2020
Nouvelles fonctions OpsCenter	OpsCenter inclut de nouvelles fonctions pour vous aider à localiser et exécuter rapidement les runbooks Automation afin de résoudre les problèmes. Pour de plus amples informations, consultez Fonctions des runbooks Automation dans OpsCenter.	19 août 2020
Nouvelle source de données dans Explorer : AWS Support cas	Exploreraffiche désormais des informations sur AWS Support les cas. Vous devez disposer d'un compte Enterprise ou Business configuré avec AWS Support. Pour de plus amples informations, consultez Modification des sources de données de Systems Manager Explorer.	13 août 2020

Distributor fournit désormais un package tiers de Trend Micro.	Distributor inclut 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 applicati ons dans le cloud. Pour plus d'informations, consultez AWSDistributor.	12 août 2020
Le plugin du document `aws:configurePackage` inclut désormais le paramètre additionalArguments.	Le plugin du document de commande Systems Manager `aws:configurePackage` prend désormais en charge la fourniture de paramètres supplémentaires à vos scripts (installation, désinstallation et mise à jour) avec le nouveau paramètre `additiona lArguments` . Pour de plus amples informations, consultez la rubrique `aws:confi gurePackage` .	11 août 2020

Contenu AppConfig déplacé dans un guide de l'utilisateur distinct	Les informations sur AWS AppConfig ont été transférées dans un guide de l'utilisateur distinct. Pour plus d'informations, voir Qu'est-ce que c'est AWSAppConfig ? AppConfig dispose également d'une page d'accueil de documentation séparée avec des liens vers le guide de l'utilisateur, la référence de l'AppConfigAPI et un nouvel AppConfig atelier.	3 août 2020
Quick Setupprend désormais en charge AWS Organizations	Quick Setupvous AWS Organizations permet désormais de configure r rapidement les rôles de sécurité requis et les fonctionn alités fréquemment utilisées de Systems Manager sur plusieurs comptes et régions. Pour plus d'informations, consultez AWS Systems ManagerQuick Setup.	23 juillet 2020
Nouvelle source de données dans Explorer : conformité des associations	Explorer affiche désormais les données de conformité des associations à partir de State Manager. Pour de plus amples informations, consultez Modification des sources de données de Systems Manager Explorer.	23 juillet 2020

Nouveau document de commande System Manager pour activer et désactiver Kernel Live Patching	Le document `AWS-ConfigureKernelLivePatching` est désormais disponible pour une utilisation avec Run Command lorsque vous voulez activer ou désactiver Kernel Live Patching sur des instances Amazon Linux 2. Ce document supprime le besoin de créer vos propres documents de commande personnalisés pour ces tâches. Pour de plus amples informations, consultez Utiliser les correctifs live du noyau sur les instances Amazon Linux 2	22 juillet 2020
Mise à jour des quotas Automation	Les quotas de service pour Automation ont été mis à jour, notamment avec une file d'attente séparée pour les automatisations de contrôle de débit. Pour plus d'informations, consultez AWS Systems Manager Automation.	20 juillet 2020

Spécifier le nombre de jours de décalage programmé pour une fenêtre de maintenance à l'aide de la console	À l'aide de la console Systems Manager, 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 n'était disponible que lors de l'utilisation d'un AWS SDK ou d'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 horaire de 2, la fenêtre ne s'exécutera que deux jours plus tard à 23h30. Pour de plus amples informations, consultez Cron et expressions de taux pour Systems Manager et Spécifier le nombre de jours de décalage programmé pour une fenêtre de maintenance.	17 juillet 2020

Mettre à jour PowerShell en utilisant Run Command	Pour vous aider à passer PowerShell à la version 5.1 sur vos instances Windows Server 2012 et 2012 R2, nous avons ajouté une procédure pas à pas dans le guide de l' AWS Systems Manager utilisateur. Pour plus d'informations, voir Mettre à jour à PowerShell l'aide deRun Command.	30 juin 2020
Patch Manager prend désormais en charge CentOS 8.0 et 8.1	Vous pouvez désormais utiliser Patch Manager pour corriger les instances CentOS 8.0 et 8.1. Pour plus d'informations, consultez les rubriques suivantes : • Sélection des correctifs de sécurité • Installation des correctifs • Fonctionnement des règles de référentiel de correctifs sur CentOS • Installation manuelle de SSM Agent sur les instances CentOS • Comment installer le SSM Agent sur des nœuds Linux hybrides	27 juin 2020

AppConfigs'intègre à AWS CodePipeline	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 rapides et fiables des applications et de l'infrastructure. CodePipelineautomatise les phases de création, de test et de déploiement de votre processus de publication chaque fois qu'un changement de code est effectué, en fonction du modèle de version que vous définissez. L'intégration de AppConfig avec CodePipeline offre les avantages suivants. Pour plus d'informations, consultez la section AppConfigIntégration avec CodePipeline. • Les clients qui géraient l'CodePipeline orchestration disposent désormais d'un moyen léger de déployer des modifications de configuration dans leurs applications sans avoir à déployer l'intégralité de leur base de code. • Les clients désireux d'utiliser AppConfig pour gérer les	25 juin 2020

	déploiements de configuration, mais qui sont limités car AppConfig ne prend pas en charge leur code ou leur magasin de configuration actuel, disposent maintenant d'options supplémentaires. CodePipeline prend en charge AWS CodeCommit GitHub, et BitBucket (pour n'en nommer que quelques-uns).	
Nouveau chapitre : Intégrations de produits et services	Pour vous aider à comprendre comment Systems Manager s'intègre aux Services AWS autres produits et services, un nouveau chapitre a été ajouté au guide de l' AWS Systems Manager utilisateur. Pour de plus amples informations, consultez Intégrations de produits et services à Systems Manager.	23 Juin 2020

Réorganisation du chapitre Automation	Pour vous faciliter, nous avons réorganisé les rubriques du chapitre Automation dans le Guide de l'utilisateur AWS Systems Manager . Par exemple, les références des actions Automation et des runbooks Automation sont devenues des sections de niveau supérieur du chapitre. Pour plus d'informations, consultez AWS Systems Manager Automation.	23 Juin 2020

Spécifier le nombre de jours de décalage de planification pour une fenêtre de maintenance	À l'aide d'un outil de ligne de commande ou d'un AWS SDK, vous pouvez désormais spécifier le 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, consultez Cron et expressions de taux pour Systems Manager et Spécifier le nombre de jours de décalage programmé pour une fenêtre de maintenance.	19 juin 2020

Prise en charge de Patch Manager pour le Kernel Live Patch sur les instances Amazon Linux 2	Les correctifs live du noyay 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 maintenant activer la fonctionnalité et appliquer des correctifs live du noyau à l'aide de Patch Manager. Pour de plus amples informations, consultez Utiliser les correctifs live du noyau sur les instances Amazon Linux 2.	16 juin 2020
Patch Manager augmente la prise en charge des versions Oracle Linux	Auparavant, Patch Manager ne prenait en charge que la version 7.6 d'Oracle Linux. Comme indiqué dans les Patch Managerconditions préalables de , la prise en charge couvre désormais les versions 7.5-7.8.	16 juin 2020

Exemple de scénario d'utilisation du paramètre `InstallOverrideList` dans les opérations d'application de correctifs	La nouvelle rubrique Exemple de scénario d'utilisation du paramètre `InstallOverrideList` décrit une politique permettant d'utiliser le paramètre `InstallOverrideList` dans le document `AWS-RunPatchBaseline` pour appliquer différents types de correctifs à un groupe cible, selon des calendriers de fenêtre de maintenance différents, tout en utilisant un seul ligne de base du patch.	11 juin 2020
Stratégies de déploiement prédéfinies pour AppConfig	AppConfig propose désormais des stratégies de déploiement prédéfinies. Pour de plus amples informations, consultez Création d'un package de déploiement.	10 juin 2020

Patch Manager prend désormais en charge Red Hat Enterprise Linux (RHEL) 7.8-8.2	Vous pouvez maintenant utiliser Patch Manager pour appliquer des correctifs aux instances RHEL 7.8 à 8.2. Pour plus d'informations, consultez les rubriques suivantes : • Sélection des correctifs de sécurité • Installation des correctifs • Fonctionnement des règles de référentiel de correctifs sur RHEL • Installation manuelle de l'SSM Agent sur les instances Red Hat Enterprise Linux • Comment installer le SSM Agent sur des nœuds Linux hybrides	9 juin 2020

Explorer prend en charge l'administration déléguée	Si vous agrégez Explorer des données provenant de plusieurs sources Régions AWS et en Comptes AWS utili sant la synchronisation des données des ressources avec AWS Organizations, nous vous suggérons de configure r un administrateur délégué pourExplorer. Un administr ateur délégué améliore la sécurité Explorer en limitant le nombre d'administrateurs Explorer qui peuvent créer ou supprimer des synchroni sations de données de ressources de plusieurs comptes et régions à une seule personne. Vous n'avez plus besoin d'être connecté au compte de gestion AWS Organizations pour administr er la synchronisation des données de ressource dans Explorer. Pour de plus amples informations, consultez Configuration d'un administr ateur délégué.	3 juin 2020

Appliquer l'association State Manager uniquement à l'intervalle Cron spécifié suivant	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 plus d'informations, consultez Création d'associations.	3 juin 2020
Nouvelle source de données dans Explorer : AWS Compute Optimizer	Exploreraffiche 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 plus d'informations, consultez les détails de configuration AWS Compute Optimizer dans Configuration des services associés.	26 mai 2020

Nouveau chapitre : balisage des ressources Systems Manager	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 • Paramètres • Références de correctifs	25 mai 2020

Installation des Service Packs Windows et des mises à niveau des versions mineures de Linux à l'aide de Patch Manager	La nouvelle rubrique Didacticiel : créer un référentiel de correctifs pour l'installation des Service Packs Windows (console) montre comment créer un référentiel de correctifs dédié exclusivement à l'installation des Service Packs Windows. La rubrique Créer un référentiel 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.	21 mai 2020
Réorganisation du chapitre sur Parameter Store	Toutes les rubriques traitant de la configuration ou de la définition des options pour les opérations Parameter Store ont été regroupées dans la section Configuration de 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.	18 mai 2020

Nouvelle rubrique pour créer des chaînes de date et d'heure pour interagir avec les opérations d'API Systems Manager.	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 opérations d'API Systems Manager.	13 mai 2020
À propos des autorisations pour le chiffrement des paramètres SecureString	La nouvelle rubrique Restreindre l'accès aux paramètres de Systems Manager à l'aide de politiques IAM explique la différence entre le chiffrement de vos `SecureString` paramètres à l'aide d'un AWS KMS key et celui Clé gérée par AWS fourni par. AWS	13 mai 2020

Patch Manager prend désormais en charge les systèmes d'exploitation Debian Server et Oracle Linux 7.6	Vous pouvez désormais utiliser Patch Manager pour appliquer le correctif aux instances Debian Server et Oracle Linux. Patch Managerprend en charge les correctifs pour Debian Server versions 8.x et 9.x et Oracle Linux 7.6. Pour plus d'informations, consultez les rubriques suivantes : • Sélection des correctifs de sécurité • Installation des correctifs • Fonctionnement des règles de référentiel de correctifs sur Debian Server • Fonctionnement des règles de référentiel de correctifs sur Oracle Linux	7 mai 2020
Créez des State Manager associations qui ciblent AWS Resource Groups	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 AWS Resource Groups. Pour de plus amples informations, consultez A propos des cibles et des contrôles du débit dans les associations State Manager	7 mai 2020

Nouveau type de données `aws:ec2:image` dans Parameter Store pour valider les ID d'AMI	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'Amazon Machine Image (AMI) 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 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 de plus amples informations, consultez Prise en charge des	5 mai 2020

	paramètres natifs pour les ID d'Amazon Machine Image.	
Gestion des codes de sortie dans les commandes Run Command	Run Command 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'ensembl e 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 Run Command.	5 mai 2020

Nouveaux paramètres publics publiés pour les zones de disponibilité et les zones locales	Des paramètres publics ont été publiés pour rendre les informations sur les zones de disponibilité et les zones locales AWS disponibles par programmation. Ils s'ajoutent aux paramètres publics de l'infrastructure mondiale existants pour Services AWS et Régions AWS. Pour plus d'informations, consultez la section Appeler des paramètres publics pour les régions Services AWS, les points de terminaison, les zones de disponibilité, les zones locales et les zones de longueur d'onde.	4 mai 2020
Nouvelle source de données dans Explorer : AWS Trusted Advisor	Exploreraffiche 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 quotas de service. Pour plus d'informations, consultez les détails de configuration Trusted Advisor dans Configuration des services associés.	4 mai 2020

Créez des State Manager associations qui exécutent Chef des recettes	Vous pouvez créer des State Manager associations qui exécutent des livres de recettes et Chef des livres de recettes à l'aide du `AWS-ApplyChefRecipes` document. Ce document offre les avantages suivants pour les Chef recettes de course à pied : • Supporte plusieurs versions de Chef (Chef11 à Chef 14). • Installe automatiquement le logiciel Chef client sur les instances cibles. • Exécute éventuellement des contrôles de conformit é 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 personnal	19 mars 2020

	isés aux exécutions `chef-client` . Pour plus d'informations, voir Création d'associations qui exécutent Chef des recettes	
Synchronisation des données d'inventaire provenant de plusieurs compartiments Comptes AWS vers un compartiment Amazon S3 central	Vous pouvez synchroniser les données d'inventaire de Systems Manager depuis plusieurs Comptes AWS compartiments vers un compartiment S3 central. Les comptes doivent être définis dans AWS Organizations. Pour de plus amples informations, consultez Création d'une synchronisation des données de ressources d'inventaire pour plusieurs comptes définis dans des organisations AWS Organizations.	16 mars 2020

Stockage de configurations AppConfig dans Amazon S3	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, consultez À propos des configurations stockées dans Amazon S3.	13 mars 2020
SSM Agent installé par défaut sur les AMIs optimisées pour Amazon ECS	SSM Agent est maintenant installé par défaut sur les AMIs optimisées pour Amazon ECS. Pour plus d'informations, consultez Utilisation de SSM Agent.	25 février 2020
Créer des configurations AppConfig dans la console	AppConfig vous permet désormais de créer une configuration d'application dans la console au moment de la création d'un profil de configuration. Pour plus d'informations, consultez Création d'une configuration et d'un profil de configuration.	13 février 2020

Approuver automatiquement uniquement les correctifs publiés jusqu'à une date spécifiée	Outre l'option permettant d'approuver automatiquement les correctifs pour l'installation un nombre spécifié de jours après leur publication, Patch Manager prend désormais en charge la possibilité d'approuver automatiquement uniquement les correctifs publiés à une date spécifiée ou avant. Par exemple, si vous spécifiez le 7 juillet 2020 comme date limite dans votre référentiel de correctifs, aucun correctif publié à partir du 8 juillet 2020 n'est installé automatiquement. Pour plus d'informations, consultez À propos des références personnalisées et Utilisation de référentiels de correctifs personnalisés.	12 février 2020

Utiliser le pseudo-paramètre {{RESOURCE_ID}} dans les tâches de la fenêtre de maintenance	Lorsque vous enregistrez une tâche de fenêtre de maintenance, vous spécifiez les paramètres uniques au type de tâche. Vous pouvez également référence r certaines valeurs en utilisant la syntaxe du pseudo-pa ramètre, comme `{{TARGET_ID}}` , `{{TARGET_TYPE}}` et `{{WINDOW_TARGET_ID}}` . Une fois que la tâche de la fenêtre de maintenan ce 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 de plus amples informati ons, consultez les rubriques suivantes dans Didacticiel : Créer et configurer une fenêtre de maintenance (AWS CLI) : • Utilisation de pseudo-pa ramètres lors de l'enregis	6 février 2020

	trement des tâches de la fenêtre de maintenance • Exemples : Enregistrement de tâches avec une fenêtre de maintenance	
Réexécuter rapidement les commandes	Systems Manager inclut deux options pour vous aider à réexécuter une commande depuis la Run Commandpage de la AWS Systems Manager console. Réexécuter : ce bouton vous permet d'exécuter la même commande sans y apporter de modifications. Copier vers 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, consultez Réexécution des commandes.	5 février 2020

Revenir du niveau des instances avancées au niveau des instances standard	Si vous avez précédemment configuré toutes les instances locales exécutées dans votre environnement hybride pour utiliser le niveau des instances avancées, vous pouvez désormais configure r 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 réunies en une Compte AWS seule instance. Région AWS Le retour au niveau des instances standard a une incidence sur la disponibilité de certaines fonctionnalités de Systems Manager. Pour de plus amples informations, consultez Retour du niveau des instances avancées au niveau des instances standard	16 janvier 2020

Nouvelle option pour ignorer les redémarrages d'instance après l'installation du correctif	Auparavant, les instances gérées étaient toujours redémarrées après que Patch Manager y avait installé des correctifs. 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, voir Nom du paramètre : RebootOption dans la rubrique À propos du document `AWS-RunPatchBaseline` SSM.	15 janvier 2020
Nouveau sujet : « Exécution de PowerShell scripts sur des instances Linux »	Une nouvelle rubrique qui décrit comment exécuter Run Command des PowerShell scripts sur des instances Linux. Pour plus d'informations, consultez Exécution de PowerShell scripts sur des instances Linux.	10 janvier 2020

Mises à jour de « Configure r SSM Agent afin d'utiliser un proxy »	Les valeurs à spécifier lors de la configuration de SSM Agent pour utiliser un proxy ont été mises à jour pour refléter les options des serveurs proxy HTTP et HTTPS. Pour de plus amples informations, consultez Configurer SSM Agent afin d'utiliser un proxy.	9 janvier 2020

Nouveau chapitre « Sécurité » décrivant les pratiques de sécurisation des ressources Systems Manager	Un nouveau chapitre Securité dans le AWS Systems Manager Guide de l'utilisateur vous permet de 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 apprendrez également à en utiliser d'autres Services AWS qui vous aident à surveiller et à sécuriser les ressources de vos 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.	24 décembre 2019

Nouveaux exemples de runbooks Automation personnalisés	Un ensemble d'exemples de runbooks 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 runbooks Automation personnalisés. Pour de plus amples informations, consultez Exemples de runbooks Automation personnalisés. Vous pouvez également afficher le contenu des runbooks Automation gérés par Amazon dans la console Systems Manager. Pour de plus amples informations, consultez la Référence des runbooks Automation de Systems Manager.	23 décembre 2019

Prise en charge d'Oracle Linux	Systems Manager prend désormais en charge Oracle Linux 7.5 et 7.7. Pour de plus amples informations sur l'installation manuelle de l'SSM Agent sur les instances EC2 pour les instances Oracle Linux, consultez Oracle Linux. Pour plus d'informations SSM Agent sur l'installation sur des Oracle Linux serveurs dans un environnement hybride, consultez Comment installer l'agent SSM sur des nœuds Linux hybrides.	19 décembre 2019

Lancement de sessions Session Manager à partir de la console Amazon EC2	Vous pouvez maintenant 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 Session Manager console 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 - Politiques IAM par défaut pour Session Manager • Démarrage d'une session (console Amazon EC2)	18 décembre 2019

CloudWatch prise en charge des Run Command métriques et des alarmes	AWS Systems Manager publie désormais des statistiques sur l'état des Run Command commandes CloudWatch, ce qui vous permet de définir des alarmes en fonction de ces mesures. Les valeurs de statut du terminal pour les commandes pour lesquelles vous pouvez suivre les métriques incluent `Success`, `Failed` et `Delivery Timed Out`. Pour plus d'informations, consultez la section Surveillance des Run Command métriques à l'aide d'Amazon CloudWatch.	17 décembre 2019

Nouvelle fonctionnalité de Systems Manager : Change Calendar	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 runbooks Systems Manager Automation ou de fonctions AWS Lambda) dans les ressources. Un document Change Calendar est un nouveau type de document Systems Manager qui stocke les données iCalendar 2.0 au format texte brut. Pour de plus amples informations, consultez Modifier le calendrier AWS Systems Manager.	11 décembre 2019

Nouvelle fonctionnalité de Systems Manager : AWSAppConfig	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 l'utiliser AppConfig avec des applications hébergées sur des instances EC2 AWS Lambda, des conteneurs, des applications mobiles ou des appareils IoT. 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 plus d'informations, consultez AWSAppConfig.	25 novembre 2019

Nouvelle fonctionnalité de Systems Manager : Systems Manager Explorer	AWS Systems Manager Explorer est un tableau de bord des opérations personnalisable qui fournit des informations sur vos AWS ressources. Exploreraffiche une vue agrégée des données d'exploitation (OpsData) pour vous Comptes AWS et pour l'ensemble de celles-ci Régions AWS. DansExplorer, OpsData inclut les métadonné es relatives à vos instances EC2, les détails de conformité des correctifs et les éléments de travail opérationnels (OpsItems). Explorerfournit un contexte sur la manière dont elles OpsItems sont réparties entre vos unités commercia les ou vos applications, sur leur évolution dans le temps et sur leur variation par catégorie . Vous pouvez regrouper et filtrer les informations dans Explorer pour vous concentre r sur les éléments qui vous intéressent et qui nécessite nt une action. Lorsque vous identifiez des problèmes prioritaires, vous pouvez utiliser la fonction OpsCenter de Systems Manager pour exécuter des runbooks Automation et résoudre rapidement ces problèmes	18 novembre 2019

. Pour plus d'informations, consultez AWS Systems ManagerExplorer.

> **Note**
>
> La configuration de Systems Manager OpsCenter est intégrée à la configuration de 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, consultez Démarrer avec Explorer et OpsCenter.

Amélioration des fonctionnalités de recherche de paramètres	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 selon `equals` et `begins-with` . Pour de plus amples informations, consultez Recherche de paramètres Systems Manager.	15 novembre 2019

Nouveau Document Builder for Automation basé sur la console | Prise en charge de l'exécution de scripts dans les étapes Automation	Vous pouvez désormais utiliser Systems Manager Automation pour créer et partager des playbooks opérationnels standardisés afin de garantir la cohérence entre les utilisateurs Comptes AWS, et Régions AWS. Grâce à la possibilité d'exécuter des scripts et d'ajouter de la documentation en ligne à vos runbooks 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. • Démonstration : Utiliser Document Builder pour créer un runbook Automation personnalisé • `aws:executeScript` (référence sur les actions Automation) • Création de runbooks Automation à l'aide de Document Builder	14 novembre 2019

	• New Automation Features In Systems Manager sur AWS News Blog	
Effectuer une mise à jour de package sur place à l'aide de Distributor	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	11 novembre 2019

Nouvelle fonction de mise à jour automatique SSM Agent	En un clic, vous pouvez configurer toutes les instances de votre Compte AWS ordinateur pour rechercher et télécharger automatiquement les nouvelles versions deSSM Agent. Pour ce faire, choisissez la mise à jour automatique de l'agent sur la page Instances gérées de la AWS Systems Manager console. Pour de plus amples informations, consultez Automatiser les mises à jour vers SSM Agent.	5 novembre 2019
Restreindre Session Manager l'accès à l'aide AWS de balises fournies	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 politiques 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 AWS fournies permet aux organisations qui utilisent des identifiants fédérés de contrôler l'accès des utilisateurs aux sessions. Pour de plus amples informations, consultez Autoriser un utilisateur à interrompre uniquement les sessions qu'il a démarrées.	2 octobre 2019

Nouveau document de commande SSM pour appliquer les Playbooks Ansible	Vous pouvez créer des State Manager associations qui exécutent Ansible des Playbooks à l'aide du `AWS-ApplyAnsiblePlaybooks` document. Ce document offre les avantages suivants pour l'exécution de manuels stratégiques : • Prise en charge de l'exécution de manuels stratégiques complexes • Support pour le téléchargement de Playbooks depuis GitHub 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 plus d'informations, voir Création d'associations qui exécutent des Ansible playbooks	24 septembre 2019

Prise en charge du réacheminement de port pour Session Manager	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 plus d'informations, consultez les rubriques suivantes : • Port Forwarding Using AWS Systems ManagerSession Manager sur AWS News Blog • Démarrage d'une session (réacheminement de port)	29 août 2019

Spécifiez un niveau de paramètre par défaut ou automatisez la sélection de niveau	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 politique de paramètre est associée au paramètre ou si les 10 000 paramètres maximum pris en charge pour le niveau standard sont déjà créés.) Pour de plus amples informations sur la spécification d'un niveau par défaut et l'utilisation d'Intelligent-Tiering, consultez Spécification d'un niveau de paramètre par défaut.	27 août 2019

Section « Travailler avec les associations » mise à jour avec la CLI et PowerShell les procédures	La section Travailler avec les associations a été mise à jour pour inclure la documentation procédurale pour la gestion des associations à l'aide du AWS CLI or AWS Tools for PowerShell. Pour de plus amples informations, consultez Utilisation d'associations dans Systems Manager.	26 août 2019
Section « Travailler avec les exécutions automatisées » mise à jour avec la CLI et PowerShell les procédures	La section Travailler avec les exécutions automatisées a été mise à jour pour inclure la documentation procédura le pour exécuter des flux de travail d'automatisation à l'aide du AWS CLI ou AWS Tools for PowerShell. Pour de plus amples informations, consultez Utilisation des exécutions Automation.	20 août 2019

OpsCenter est intégré à Application Insights	OpsCenters'intègre 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 plus d'informations sur la configuration d'Application Insights pour créerOpsItems, consultez la section Configurer, configurer et gérer votre application à des fins de surveillance dans le guide de CloudWatch l'utilisateur Amazon.	7 août 2019

Nouvelle fonctionnalité de console : AWS Systems ManagerQuick Setup	Quick Setup est une nouvelle fonction de la console Systems Manager qui vous permet de configure r rapidement plusieurs composants Systems Manager sur vos instances EC2. Plus précisément, la configuration rapide vous aide à configure r les composants suivants sur les instances que vous sélectionnez ou ciblez à l'aide de balises : • Rôle de profil d'instance AWS Identity and Access Management (IAM) pour Systems Manager. • Une mise à jour planifiée bimensuelle d'SSM Agent. • Une collecte planifiée de métadonnées d'inventaire toutes les 30 minutes. • Une analyse quotidienne de vos instances pour identifier les correctifs manquants. • Installation et configuration uniques de l' CloudWatc hagent Amazon. • Une mise à jour mensuelle planifiée de l' CloudWatch agent. Pour de plus amples informati ons, consultez Configura	7 août 2019

tion rapide AWS Systems Manager.

Enregistrer un groupe de ressources en tant que cible d'une fenêtre de maintenance	Outre l'enregistrement des instances gérées en tant que cible d'une fenêtre de maintenance, vous pouvez désormais enregistrer un groupe de ressource s en tant que cible d'une fenêtre de maintenance. Maintenance Windowspr end en charge tous les types de AWS ressources pris en charge en AWS Resource Groups incluant`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 Run Command console ou de la AWS CLI send-command commande. Pour plus d'informations, consultez les rubriques suivantes : • Affecter des cibles à une fenêtre de maintenance (console) • Exemples : Enregistrement de cibles avec une fenêtre de maintenance	23 juillet 2019

	• Utilisation des contrôles de cibles et de taux pour envoyer des commandes à un parc	
Création de packages et gestion des versions simplifiées avec AWS Systems ManagerDistributor	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.	22 juillet 2019
Nouveau panneau de catégories de document pour Systems Manager Automation	Systems Manager inclut un nouveau panneau de catégories de document lorsque vous exécutez une automatisation dans la console. Utilisez ce panneau pour filtrer les runbooks d'Automation en fonction de leur finalité.	18 juillet 2019

Vérification des autorisations utilisateur pour l'accès au document de configuration Session Manager par défaut	Lorsqu'un utilisateur de votre compte utilise le AWS CLI pour démarrer une Session Manager session sans spécifier de document de configuration dans la commande, Systems Manager utilise le document de configuration par défaut`SSM-SessionManagerRunShell` . Vous pouvez désormais vérifier que l'utilisateur a reçu l'autorisation d'accéder à ce document en ajoutant un élément de condition pour `ssm:SessionDocumentAccessCheck` à la politique du.AWS Identity and Access Management entité (IAM) (utilisateur, groupe ou rôle). Pour de plus amples informations, consultez Application de la vérification d'autorisation de document pour le scénario d'interface de ligne de commande par défaut.	9 juillet 2019

Prise en charge du démarrage de sessions Session Manager à l'aide des informations d'identification de l'utilisateur de système d'exploitation	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, consultez Activer la prise en charge de Run As pour les instances Linux.	9 juillet 2019
Prise en charge du démarrage de sessions Session Manager à l'aide de SSH	Vous pouvez désormais utiliser le AWS CLI pour démarrer une session SSH sur une instance gérée en utilisantSession Manager. Pour de plus amples informations sur l'activation de sessions SSH avec Session Manager, consultez (Facultatif) Activer des sessions SSH Session Manager. Pour de plus amples informations sur le lancement d'une session SSH avec Session Manager, consultez Démarrage d'une session (SSH).	9 juillet 2019

Prise en charge de la modification des mots de passe sur les instances gérées	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, consultez Réinitialisation de mots de passe sur des instances gérées.	9 juillet 2019
Révisions apportées à « Qu'est-ce que c'est AWS Systems Manager ? »	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 de 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.	10 juin 2019

Nouvelle fonctionnalité de Systems Manager : OpsCenter	OpsCenterfournit un emplacement central où les ingénieurs des opérations et les professionnels de l'informatique peuvent consulter, étudier et résoudre les éléments de travail opérationnels (OpsItems) liés aux AWS ressources. OpsCenter est conçu pour réduire le délai moyen de résolution des problèmes ayant une incidence sur les AWS ressources. 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 runbooks Automation Systems Manager 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’informa	6 juin 2019

	tions, consultez AWS Systems ManagerOpsCenter.	
Modifications apportées au volet de navigation gauche de Systems Manager dans AWS Management Console	Le volet de navigation gauche de Systems Manager AWS Management Console inclut de nouveaux en-têtes, dont un nouveau titre pour Ops Center, qui fournissent un regroupement plus logique des fonctionnalités de Systems Manager.	6 juin 2019

Révision du didacticiel pour la création et la configuration d'une fenêtre de maintenance à l'aide de l' AWS CLI	Le didacticiel Création et configuration d'une fenêtre de maintenance (AWS CLI) a été révisé afin de fournir un parcours simple 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'enregis trement des tâches, y compris les informations d'utilisation des pseudo-paramètres tels que `{{TARGET_ID}}` . Pour obtenir plus d'informations et d'autres exemples, consultez les rubriques suivantes : • Exemples : Enregistrement de cibles avec une fenêtre de maintenance • Exemples : Enregistrement de tâches avec une fenêtre de maintenance • À propos des register-task-with-maintenance options - windows • Utilisation de pseudo-pa ramètres lors de l'enregis	31 mai 2019

	trement des tâches de la fenêtre de maintenance	
Notifications sur les mises à jour de SSM Agent	Pour être informé des SSM Agent mises à jour, abonnez-vous à la page des notes de SSM Agent publication surGitHub.	24 mai 2019
Réception de notifications ou déclenchement d'actions en fonction des modifications apportées à Parameter Store	La rubrique Configurer des notifications ou déclencher des actions en fonction d'Parameter Storeévénements vous permet désormais de configurer EventBridge les règles Amazon pour répondre aux modifications apportées 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.	22 mai 2019

Révisions majeures du contenu des chapitres Configuration et Mise en route	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 configura tion 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 virtuelle s dans un environnement hybride. Les deux sections contiennent désormais toutes les rubriques de configura tion sous la forme d'étapes principales numérotées, classées par ordre d'achèvem ent recommandé. Le nouveau chapitre Mise en route a pour objectif d'aider les utilisate urs finaux à démarrer avec Systems Manager après avoir terminé les tâches de configuration de compte et de service. • Configuration AWS Systems Manager	15 mai 2019

- Configuration AWS Systems Manager pour les environnements hybrides
- Commencer avec AWS Systems Manager

Inclusion de correctifs pour les applications publiées par Microsoft dans les référentiels de correctifs (Windows)	Patch Manager prend désormais en charge les mises à jour de correctifs pour les applications publiées par Microsoft sur les instances Windows Server. Auparavant, seuls les correctifs pour le système d'exploitation Windows Server étaient pris en charge. Patch Manager fournit deux référentiels de correctifs prédéfinis pour des instances Windows Server. La référentiel de correctifs `AWS-WindowsPredefinedPatchBaseline-OS` s'applique aux correctifs de système d'exploitation uniquement. `AWS-WindowsPredefinedPatchBaseline-OS-Applications` s'applique à la fois le système d'exploitation Windows Server et les applications publiées par Microsoft sur Windows. Pour de plus amples informations sur la création d'un référentiel de correctifs personnalisée qui inclut des correctifs pour les applications publiées par Microsoft, consultez la première procédure dans Création d'un référentiel de correctifs personnalisée. Dans le cadre de cette mise à	7 mai 2019

	jour, les noms des lignes de base de correctifs prédéfinies AWS fournies sont également modifiés. Pour de plus amples d'informations, consultez Références prédéfinies.	
Exemples d'enregistrement des cibles de fenêtre de maintenance à l'aide du AWS CLI	La nouvelle rubrique Exemples : Enregistrement de cibles avec une fenêtre de maintenance fournit trois exemples de commandes qui illustrent les différentes méthodes de spécification des 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.	3 mai 2019

Mises à jour des rubriques de groupes de correctifs	La rubrique À propos des groupes de correctifs a été mise à jour pour inclure une section sur la façon dont les instances gérées détermine nt le référentiel de correctifs appropriée lors des opération s d'application de correctif s. En outre, des instructi ons ont été ajoutées pour utiliser la console AWS CLI ou Systems Manager pour ajouter un groupe de correctif s ou des PatchGroupbalises à vos instances gérées, et pour savoir comment ajouter un groupe de correctifs ou PatchGroupà une ligne de base de correctifs. (Utilisez **`PatchGroup`** , sans espace, si vous avez autorisé les balises dans les métadonnées d'instance EC2.. Pour de plus amples informations, consultez Créer un Groupe de correctif s et Ajouter un Groupe de correctifs pour un référentiel de correctifs.	1er mai 2019

Nouvelles fonctions Parameter Store	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 un quota de taille plus élevé pour la valeur du paramètre, un quota plus élevé pour le nombre de paramètres que vous pouvez créer par Compte AWS et Région AWS la possibilité d'utiliser des politiques de paramètres. Pour de plus amples informations sur les paramètres avancés, consultez la section À propos des paramètres avancés Systems Manager. • Paramètre de politiques : les politiques de paramètre vous aident à gérer un ensemble croissant de paramètres en vous permettant d'attribuer des critères spécifiques à un paramètre, par exemple	25 avril 2019

une date d'expiration ou une durée de vie. Les politiques 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 politiques de paramètre ne sont disponibles que pour les paramètres qui utilisent le niveau entre les paramètres avancés. Pour de plus amples informations, consultez Utilisation des politiques de paramètres.

- Débit supérieur : vous pouvez désormais augmenter le quota de débit Parameter Store à un maximum de 1 000 transactions par seconde. Pour de plus amples informations, consultez la section Accroissement du débit Parameter Store.

Mise à jour de la section relative à Automation	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 Automation : • Exécution manuelle d'une automatisation • Exécution d'une automatisation avec des approbateurs • Planification des automatisations	17 avril 2019

Chiffrer les données de session à l'aide d'une clé AWS KMS	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 maintenant choisir de chiffrer davantage ces données à l'aide AWS KMS key d'un code créé dans AWS Key Management Service. Vous pouvez utiliser une clé KMS qui a été créée dans votre Compte AWS ou une clé qui a été partagée avec vous à partir d'un autre compte. Pour plus d'informations sur la spécification d'une clé KMS pour chiffrer les données de session, voir Activer le chiffrement par AWS KMS clé des données de session (console), Créer des Session Manager préférences (AWS CLI) ou Mettre à jour les Session Manager préférences (AWS CLI).	4 avril 2019

Configuration des notifications Amazon SNS pour AWS Systems Manager	Ajout d'instructions pour utiliser la console AWS CLI ou Systems Manager afin de configurer les notifications Amazon SNS Run Command et les Run Command tâches enregistrées dans une fenêtre de maintenance. Pour plus d'informations, consultez Configuration des notifications Amazon SNS pour AWS Systems Manager.	6 mars 2019

Instances avancées pour les serveurs et les machines virtuelles dans des environnements hybrides	AWS Systems Manager propose un niveau d'instances standard et un niveau d'instances avancées pour les serveurs et les machines virtuelles de votre environnement hybride. Le niveau des instances standard vous permet d'enregistrer un maximum de 1 000 serveurs ou machines virtuelles par personne. Compte AWS 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 disponibles sur une base. pay-per-use Les instances avancées vous permettent également de vous connecter à vos machines hybrides en utilisant AWS Systems Manager Session Manager. Session Managerfournit un accès shell interactif à vos instances. Pour de plus amples informations sur l'activation des instances	4 mars 2019

	avancées, consultez Utilisation du niveau des instances avancées.	
Créer des associations State Manager qui utilisent des documents SSM partagés	Vous pouvez créer des State Manager associations qui utilisent des runbooks SSM Command and Automation partagés avec d'autres utilisateurs. Comptes AWS La création d'associations à l'aide de documents SSM partagés permet de conserver votre instance 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, consultez Créer une association.	28 février 2019
Afficher les listes des événements Systems Manager pris en charge par les EventBridge règles Amazon	La nouvelle rubrique Monitoring Systems Manager events with Amazon EventBridge fournit un résumé des différents événements émis par Systems Manager pour lesquels vous pouvez configurer des règles de surveillance des événements EventBridge.	25 février 2019

Ajouter des balises lorsque vous créez des ressources Systems Manager	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 à l'aide du SDK AWS CLI ou d'un SDK incluent les fenêtres de maintenance, les lignes de base des correctifs, Parameter Store les paramètres et les 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.	24 février 2019

Création automatique d'un rôle IAM pour Systems Manager Inventory	Auparavant, vous deviez créer un rôle AWS Identity and Access Management (IAM) et associer des politiques distinctes à ce rôle pour afficher les données d'inventaire sur la page d'affichage détaillé de l'inventaire de la console. Vous n'avez plus besoin de créer ce rôle ni d'attacher des politiques à celui-ci. Lorsque vous choisissez une synchronisation des données à distance sur la page d'affichage détaillé de l'inventaire, Systems Manager crée automatiquement le `Amazon-GlueServicePolicyForSSM` rôle et lui attribue la politique Amazon- GlueService PolicyFor SSM- {S3 bucket name} ainsi que la AWSGlueServiceRole politique correspondante. Pour de plus amples informations, consultez Interrogation des données d'inventaire à partir de plusieurs régions et comptes.	14 février 2019

Démonstrations Maintenance Windows pour mettre à jour l'SSM Agent	Ajout de deux nouvelles procédures à la documentation Maintenance Windows. Les procédures pas à pas expliquent comment utiliser la console Systems Manager ou comment AWS CLI créer une fenêtre de maintenance qui se conserve SSM Agent up-to-date automatiquement. Pour de plus amples informations, consultez Procédures Maintenance Windows.	11 février 2019
Utilisation de paramètres publics de Parameter Store	Ajout d'une brève section décrivant les paramètres Parameter Store publics. Pour de plus amples informations, consultez Utilisation des paramètres publics de Systems Manager.	31 janvier 2019
Utilisez le AWS CLI pour créer des Session Manager préférences	Des instructions ont été ajoutées pour utiliser le AWS CLI pour créer des Session Manager préférences, telles que CloudWatch les journaux, les options de journalisation du compartiment S3 et les paramètres de chiffrement de session. Pour plus d'informations, voir Utiliser le AWS CLI pour créer des Session Manager préférences.	22 janvier 2019

Exécution de flux de travail Systems Manager Automation avec State Manager	AWS Systems Manager State Managerprend désormais en charge la création d'associations utilisant les runbooks SSM Automation. State Managerau paravant uniquement pris en charge `command` et `policy` documents, ce qui signifiait que vous ne pouviez créer que des associations ciblant les instances gérées. Grâce à la prise en charge des runbooks Automation SSM, vous pouvez désormais créer des associati ons qui ciblent les différents types de ressources AWS . Pour de plus amples informati ons, consultez Exécution des flux de travail Systems Manager Automation avec State Manager.	22 janvier 2019

Mises à jour des références pour les expressions Cron et Rate et pour les options de planification de fenêtre de maintenance	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 Maintenance Windows 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.	6 décembre 2018
Activer la journalisation du débogage SSM Agent	Vous pouvez activer la journalisation de débogage de l'SSM Agent en modifiant le fichier seelog.xml .template sur l'instance gérée. Pour de plus amples informations, consultez Activer la journalisation du débogage de l'SSM Agent.	30 novembre 2018

Prise en charge des architectures de processeur ARM64	AWS Systems Manager prend désormais en charge les versions ARM64 des systèmes d'exploitation Amazon Linux 2, Red Hat Enterprise Linux 7.6 et Ubuntu Server (18.04 LTS et 16.04 LTS). 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 plus d'informations sur le type d'instance A1, consultez la section Instances à usage général dans le guide de l'utilisateur Amazon EC2.	26 novembre 2018

Créez et déployez des packages en utilisant AWS Systems ManagerDistributor	À l'aide de ce package AWS Systems Manager Distributor, vous pouvez créer votre propre package logiciel (ou rechercher des packages logiciels d'agent AWS fournis, par exemple`AmazonCloudWatchAgent`) à installer sur des instances gérées. AWS Systems Manager Distributorpublie des ressources, telles que des progiciels, sur des instances AWS Systems Manager gérées. La publication d'un package permet de publier des versions spécifiques du document du package (un document Systems Manager, que vous créez lorsque vous ajoutez le package dans Distributor) sur les instances gérées que vous identifiez grâce aux ID d'instance gérée, aux ID de Compte AWS , aux balises ou à une Région AWS. Pour plus d'informations, consultez AWS Systems ManagerDistributor.	20 novembre 2018

Exécutez simultanément des flux de travail AWS Systems Manager d'automatisation sur plusieurs comptes Régions AWS et Comptes AWS à partir d'un compte central	Vous pouvez exécuter simultanément des flux de travail AWS Systems Manager d'automatisation sur plusieurs et/ou Régions AWS unités AWS organisationnelles (UO) à partir d'un compte de gestion d'automatisation. Comptes AWS Les automatisations exécutées en simultané dans différentes régions et comptes ou unités organisationnelles permettent de réduire le temps nécessaire pour administrer vos ressources AWS , tout en renforçant la sécurité de votre environnement informatique. Pour plus d'informations, consultez la section Exécution de flux de travail d'automatisation dans plusieurs Régions AWS et Comptes AWS.	19 novembre 2018

Interrogez les données d'inventaire à partir de plusieurs Régions AWS et Comptes AWS	Systems Manager Inventory s'intègre à Amazon Athena pour vous aider à interroger les données d'inventaire provenant de plusieurs Régions AWS et. Comptes AWS L'intégration d'Athena utilise la synchronisation des données des ressources afin que vous puissiez consulter les données d'inventaire de toutes vos instances gérées sur la page d'affichage détaillé de l'inventaire de la AWS Systems Manager console. Pour de plus amples informations, consultez Interrogation des données d'inventaire à partir de plusieurs régions et comptes.	15 novembre 2018

Création d'associations State Manager qui exécutent des fichiers MOF	Vous pouvez exécuter des fichiers MOF (Managed Object Format) pour appliquer un état ciblé aux instances gérées de Windows Server avec State Manager en utilisant le document SSM `AWS-ApplyDSCMofs` . 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 ciblé 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 des fichiers de configuration MOF à partir d'Amazon Simple Storage Service (Amazon S3), d'un partage local ou d'un site Web sécurisé avec un domaine HTTPS. Pour de plus amples informations, consultez la section Création d'associations qui exécutent des fichiers MOF.	15 novembre 2018

Restreindre l'accès administratif dans les sessions Session Manager	Les sessions Session Manager sont lancées en utilisant les informations d'identification d'un compte utilisateur créé avec les privilèges racine par défaut ou les autorisations administrateur appelés `ssm-user`. Les informations sur la restriction de contrôle administratif pour ce compte sont maintenant disponibles dans la rubrique Activer ou désactiver les autorisations administratives de compte ssm-user.	13 novembre 2018
Exemples YAML dans Référence d'actions Automation	La référence d'actions d'automatisation inclut désormais un modèle YAML pour chaque action qui comprend déjà un modèle JSON.	31 octobre 2018

Attribuer des niveaux de sévérité de conformité aux associations	Vous pouvez désormais attribuer des niveaux de sévérité de conformité pour les associations State Manager. Ces niveaux de sévérité sont présentés dans le tableau de bord de conformité et peuvent également être utilisés pour filtrer vos rapports de conformité. Les niveaux de sévérité que vous pouvez attribuer incluent : Critique, Élevé, Moyen, Faible et Non précisé. Pour de plus amples informations, consultez Créer une association (console).	26 octobre 2018
Utilisation de cibles et de contrôles du débit avec Automation et State Manager	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, consultez 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.	23 octobre 2018

Spécification de plages de temps actives et de fuseaux horaires internationaux pour les fenêtres de maintenance	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, consultez Création d'une fenêtre de maintenance (console) et Mise à jour d'une fenêtre de maintenance (AWS CLI).	9 octobre 2018
Gestion d'une liste personnalisée de correctifs pour votre référentiel de correctifs dans un compartiment S3	Avec le nouveau paramètre « InstallOverride List » du document de commande SSM`AWS-RunPatchBaseline`, vous pouvez spécifier une URL https ou une URL de type chemin Amazon Simple Storage Service (Amazon S3) vers une liste de correctifs à installer. Cette liste d'installation de correctifs que vous gérez dans un compartiment S3 au format YAML remplace les correctifs spécifiés par le référentiel de correctifs par défaut. Pour plus d'informations, consultez la section Nom du paramètre : InstallOverrideList.	5 octobre 2018

Contrôle étendu si les dépendances de correctifs sont installées	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'install er ces dépendances ou de bloquer leur installation. Pour de plus amples informations, consultez Créer un référentiel de correctif.	5 octobre 2018
Créer des flux de travail Automation dynamiques avec des ramifications condition nelles	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 runbook Automation en fonction des résultats de cette évaluation. Pour plus d'informa tions, consultez la rubrique Using conditional statement s in runbooks (Utilisation d'instructions conditionnelles dans les runbooks).	26 septembre 2018

Utilisez le AWS CLI pour mettre à jour Session Manager les préférences	Les instructions d'utilisation de la CLI pour mettre à jour les Session Manager préférences, telles que CloudWatch les journaux et les options de journalisation des compartiments S3, ont été ajoutées au guide de l'AWS Systems Manager utilisateur. Pour plus d'informations, voir Utiliser le AWS CLI pour mettre à jour Session Manager les préférences.	25 septembre 2018
Mise à jour de l'exigence SSM Agent pour Session Manager	Session Manager exige désormais SSM Agent version 2.3.68.0 ou versions ultérieures. Pour plus d'informations sur les prérequis Session Manager, consultez Exécuter les opérations prérequises Session Manager.	17 septembre 2018

Gestion des instances sans ouverture des ports entrants ou maintenance des hôtes bastion utilisant Session Manager	À l'aide Session Manager d'une fonctionnalité entièrement gérée de AWS Systems Manager, vous pouvez gérer vos instances EC2 via un shell interactif basé sur un navigateur en un clic ou via le. AWS CLISession Managerfournit une gestion d'instance sécurisée et vérifiable sans qu'il soit nécessaire d'ouvrir les ports entrants, de gérer les hôtes Bastion ou de gérer les clés SSH. Session Managervous permet également de vous conformer aux politiques d'entreprise qui exigent un accès contrôlé aux instances, des pratiques de sécurité strictes et des journaux entièrement vérifiables avec les détails d'accès aux instances, tout en fournissant aux utilisateurs finaux un accès multiplateforme en un clic à vos instances EC2. Pour de plus amples informations, consultez Pour en savoir plus sur Session Manager.	11 septembre 2018

Invoquer un autre outil Services AWS depuis un flux de travail d'automatisation de Systems Manager	Vous pouvez invoquer d'autres Services AWS fonctionnalités de Systems Manager dans votre flux de travail d'automatisation en utilisant trois nouvelles actions d'automatisation (ou plugins) dans vos runbooks d'automatisation. Pour plus d'informations, consultez la rubrique Using action outputs as inputs (Utilisation de sorties d'action en tant qu'entrées).	28 août 2018
Utilisation de clés de condition spécifiques à Systems Manager dans les politiques IAM	La rubrique Spécification des conditions dans une politique a été mise à jour pour répertorier les clés de condition IAM pour Systems Manager que vous pouvez incorporer dans les politiques. Vous pouvez utiliser ces clés pour spécifier les conditions dans lesquelles une politique doit prendre effet. La rubrique inclut également des liens vers des exemples de politiques et d'autres rubriques connexes.	18 août 2018

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`. Pour de plus amples informations, consultez Agrégation des données d'inventaire.	16 août 2018
Affichage du suivi des modifications et de l'historique pour l'inventaire et la conformité de configuration	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 associations State Manager et l'application des correctifs de Patch Manager signalées par le service de conformité de configuration. Pour de plus amples informations, consultez Affichage du suivi des modifications et de l'historique d'inventaire.	9 août 2018

Parameter Store s'intègre à Secrets Manager	Parameter Storeest désormais intégré AWS Secrets Manager afin que vous puissiez récupérer les secrets de Secrets Manager lorsque vous en utilisez un autre Services AWS qui prend déjà en charge les référence s aux Parameter Store paramètres. Ces services incluent Amazon EC2, Amazon Elastic Container Service,,, AWS Lambda, AWS CloudFormation AWS CodeBuild AWS CodeDeplo y, et d'autres fonctionnalités de 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, consultez la section Référencement de AWS Secrets Manager secrets à partir de Parameter Store paramètres.	26 juillet 2018

Attacher des étiquettes aux paramètres Parameter Store	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, consultez Étiquetage des paramètres.	26 juillet 2018

Création de flux de travail Automation dynamiques	Par défaut, les étapes (ou actions) que vous définissez dans la section mainSteps d'un runbook Automation sont exécutées par ordre séquentiel. Lorsqu'une action est terminée, la prochaine action spécifiée dans la section mainSteps 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 obtenir des informations, consultez la rubrique Using conditional statements in runbooks (Utilisation d'instructions conditionnelles dans les runbooks).	18 juillet 2018

L'SSM Agent est désormais préinstallé sur les Ubuntu ServerAMIs 16.04 utilisant Snap	À partir des instances créées depuis les AMIs Ubuntu Server 16.04 identifiées par `20180627`, l'SSM Agent est pré-installé à l'aide de packages Snap. Pour les instances créées à partir d'AMIs antérieures, vous devez continuer à utiliser les packages du programme d'installation deb. Pour plus d'informations, consultez A propos des installations de l'SSM Agent sur les instances 16.04 64 bits Ubuntu Server.	7 juillet 2018

Vérification des autorisations S3 minimales requises par SSM Agent	La nouvelle rubrique Autorisations minimales relatives au compartiment S3 pour SSM Agent 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 politique 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.	5 juillet 2018
Affichage de l'historique d'exécution complet correspondant à un ID d'association State Manager spécifique	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.	2 juillet 2018

Patch Manager introduit la prise en charge d'Amazon Linux 2	Vous pouvez désormais utiliser Patch Manager pour appliquer des correctifs aux instances Amazon Linux 2. Pour des informations générales sur les systèmes d'exploitation pris en charge par Patch Manager, consultez Prérequis de Patch Manager. 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 la PatchFilterréférence des AWS Systems Manager API.	26 juin 2018
Envoyer le résultat de la commande à Amazon CloudWatch Logs	La nouvelle rubrique Configuration d'Amazon CloudWatch Logs pour Run Command décrit comment envoyer des Run Command résultats à CloudWatch Logs.	18 juin 2018

Créez ou supprimez rapidement la synchronisation des données de ressource pour l'inventaire en utilisant AWS CloudFormation	Vous pouvez l'utiliser AWS CloudFormation pour créer ou supprimer une synchronisation des données de ressource s pour Systems Manager Inventory. Pour l'utiliser AWS CloudFormation, ajoutez la ressource AWS::SSM: :ResourceDataSync à votre AWS CloudFormation modèle. Pour de plus amples informati ons, consultez Utilisation de modèles AWS CloudForm ation dans le Guide de l'utilisa teur AWS CloudFormation . Vous pouvez également créer manuellement une synchronisation des données de ressource pour l'inventaire comme décrit dans Configura tion de la synchronisation des données de ressource pour l'inventaire.	11 juin 2018

AWS Systems Manager Les notifications de mise à jour du guide de l'utilisateur sont désormais disponibles via RSS	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.	6 juin 2018
Spécifiez un code de sortie dans les scripts afin de redémarrer les instances gérées	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 Run Command.	3 juin 2018

Créez un événement sur Amazon EventBridge chaque fois que l'inventaire personnalisé est supprimé	La nouvelle rubrique Afficher les actions de suppression d'inventaire EventBridge décrit comment configurer Amazon EventBridge pour créer un événement chaque fois qu'un utilisateur supprime un inventaire personnalisé.	1er juin 2018

Mises à jour antérieures à juin 2018

Le tableau ci-après décrit des modifications importantes apportées dans chaque version du Guide de l'utilisateur AWS Systems Manager avant juin 2018.

Modification	Description	Date de publication
Répertoriez toutes les instances gérées dans votre Compte AWS	Vous pouvez inventorier toutes les instances gérées de votre inventaire Compte AWS en créant une association d'inventaire globale. Pour plus d'informations, consultez Répertoriez tous les nœuds gérés de votre Compte AWS. **Note** Les associations d'inventaire global sont disponibles dans SSM Agent, version 2.0 790.0 ou version ultérieure. Pour plus d'informations sur la mise à jour de l'SSM Agent sur vos instances, consultez Mise à jour de SSM Agent à l'aide de Run Command.	3 mai 2018
SSM Agent installé par défaut sur Ubuntu Server 18	SSM Agent est installé, par défaut, sur les AMIs Ubuntu Server 18.04 LTS 64 bits et 32 bits.	2 mai 2018

Modification	Description	Date de publication
Nouvelle rubrique	La nouvelle rubrique Exécution de commandes à l'aide d'une version de document spécifique décrit comment utiliser le paramètre document-version 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 décrit comment supprimer des données d'inventaire personnalisées à partir d'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 fonctionnalité utilise le fonctionnement de DeleteInventoryl'API.	19 avril 2018
Notifications Amazon SNS pour SSM Agent	Vous pouvez vous abonner à une rubrique Amazon SNS pour recevoir des notifications lorsqu'une nouvelle version de l'SSM Agent est disponible. Pour plus d'informations, consultez Abonnement aux notifications SSM Agent.	9 avril 2018
Prise en charge des correctifs CentOS	Systems Manager prend désormais en charge l'application de correctifs aux instances CentOS. Pour plus d'informations sur les versions de CentOS prises en charge, consultez Conditions préalables requises Patch Manager. Pour en savoir plus sur la façon dont les correctifs fonctionnent, consultez Fonctionnement des opérations Patch Manager.	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, AWS Systems Manager référence. Du contenu supplémentaire sera ajouté à cette section dès qu'il sera disponible.	15 mars 2018

Modification	Description	Date de publication
Nouvelle rubrique	La nouvelle rubrique À propos des formats de noms de package pour les listes de correctifs approuvés et rejetés détaille les formats de noms de package que vous pouvez saisir dans les listes de correctifs approuvés et rejetés pour un référentiel de correctifs personnalisée. Des exemples de formats sont fournis pour chaque type de système d'exploitation pris en charge par Patch Manager.	9 mars 2018
Nouvelle rubrique	Systems Manager s'intègre désormais à Chef Chef InSpec. InSpec est un framework d'exécution open source qui vous permet de créer des profils lisibles par l'homme sur Amazon GitHub 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 Utilisation de Chef InSpec profils avec Systems Manager Compliance.	7 mars 2018
Nouvelle rubrique	La nouvelle rubrique Utilisation des rôles liés aux services pour Systems Manager décrit comment utiliser un rôle lié à un service AWS Identity and Access Management (IAM) dans 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 Patch Manager pour installer les correctifs qui se trouvent dans un autre référentiel source que 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'un référentiel de correctifs personnal isée. Pour plus d'informations, consultez les rubriques suivantes : • Spécification d'un autre référentiel source de correctifs (Linux) • Utilisation des référentiels de correctifs personnalisés • Création d'un référentiel de correctifs avec des référenti els personnalisés pour les différentes versions du système d'exploitation En outre, vous pouvez utiliser Patch Manager pour corriger des instances SUSE Linux Enterprise Server. Patch Manager prend en charge l'application des correctifs sur les 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 • Installation des correctifs • Fonctionnement des règles de référence de correctif sur SUSE Linux Enterprise Server	6 février 2018

Modification	Description	Date de publication
Nouvelle rubrique	La nouvelle rubrique À propos des documents SSM pour l'application de correctifs aux nœuds gérés décrit les sept documents SSM disponibles pour vous aider à appliquer les dernières mises à jour de sécurité à vos instances gérées.	10 janvier 2018
Mises à jour importantes concernant la prise en charge de Linux	Mise à jour de différentes rubriques avec les informations suivantes : • SSM Agentest installé, par défaut, sur la base Amazon Linux 1 AMIs datée du 09/2017 et versions ultérieures. • Installez manuellement SSM Agent sur les autres versions de Linux, y compris les images ne faisant pas partie de la base de données, comme les AMIs optimisée s pour Amazon ECS.	9 janvier 2018
Nouvelle rubrique	Une nouvelle rubrique, À propos du document SSM `AWS-RunPatchBaseline`, explique le fonctionnement de ce document SSM sur les systèmes Windows et Linux. Elle offre également des informations concernant les deux paramètres disponibles dans le document `AWS-RunPatchBaseline`, `Operation` et `Snapshot ID`.	5 janvier 2018
Nouvelles rubriques	Une nouvelle section, Fonctionnement des opérations Patch Manager, fournit des détails techniques sur la manière dont Patch Manager détermine quels correctifs de sécurité installer et dont il 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

Modification	Description	Date de publication
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 « référence des runbooks Automation Systems Manager ». Celui-ci a été déplacé vers le nœud Ressources partagées > Documents pour être plus près de Référence de plug-in de document Command. Pour plus d'informations, consultez Référence sur les actions Systems Manager Automation.	20 décembre 2017
Nouveau chapitre et contenu sur la surveillance	Un nouveau chapitre fournit des instructions pour envoyer des métriques et des données de journal à Amazon CloudWatch Logs. Surveillance AWS Systems Manager Une nouvelle rubrique fournit des instructions pour la migration des tâches de surveillance sur instance, sur les Windows Server instances 64 bits uniquement, depuis SSM Agent l' CloudWatch agent. Envoi des journaux des nœuds vers CloudWatch des journaux unifiés (CloudWatch agent)	14 décembre 2017
Nouveau chapitre	Un nouveau chapitre fournit des informations complètes sur l'utilisation AWS Identity and Access Management (IAM) et permet de sécuriser AWS Systems Manager l'accès à vos ressources grâce à l'utilisation d'informations d'identification. Gestion des identités et des accès pour AWS Systems Manager Ces informations d'identification fournissent les autorisations requises pour accéder aux AWS ressources, telles que l'accès aux données stockées dans des compartiments S3, l'envoi de commandes et la lecture des balises sur les instances EC2.	11 décembre 2017
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

Modification	Description	Date de publication
Diverses modifications pour re:Invent 2017	• Lancement officiel de AWS Systems Manager : AWS Systems Manager (anciennement Amazon EC2 Systems Manager) est une interface unifiée qui vous permet de centraliser les données opérationnelles et d'automatiser les tâches entre AWS vos ressources. Vous pouvez accéder à la nouvelle AWS Systems Manager console ici. Pour de plus amples informations, consultez Qu'est-ce que c'est AWS Systems Manager ?. • Prise en charge de YAML : vous pouvez créer des documents SSM au format YAML. Pour plus d'informations, consultez AWS Systems Manager Documents.	29 novembre 2017
Utilisation de Run Command pour prendre des instantanés VSS de volumes EBS	À l'aide de Run Command, vous pouvez prendre des instantanés cohérents avec les applications 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 la section Prendre des instantanés compatibles Microsoft VSS à l'aide du guide de AWS Systems Manager l'utilisateur Amazon EC2.	le 20 novembre 2017

Modification	Description	Date de publication
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 de l'interface sont alimentés par PrivateLink une technologie qui vous permet d'accéder en privé aux API Amazon EC2 et Systems Manager à l'aide d'adresses IP privées. PrivateLink restreint tout le trafic réseau entre vos instances gérées, Systems Manager et EC2 vers le réseau Amazon (les instances gérées n'ont pas accès à Internet). 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 Améliorer la sécurité des instances EC2 en utilisant des points de terminaison VPC pour Systems Manager.	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 du dernier 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 fonction, état d'installation, etc. Avant de tenter de collecter des informations pour ces types d'inventaire, mettez à jour SSM Agent sur les instances à inventorier. En exécutant la dernière version de SSM Agent, 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'SSM Agent à l'aide de State Manager, consultez Démonstration : Mise à jour automatique de l'SSM Agent (CLI). Pour en savoir plus sur Inventory, consultez En savoir plus sur Systems Manager Inventory.	le 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 Configuration d'Automation.	31 octobre 2017

Modification	Description	Date de publication
GitHubet intégration avec Amazon S3	Exécuter des scripts à distance : Systems Manager prend désormais en charge le téléchargement et l'exécution de scripts depuis un GitHub référentiel privé ou public et depuis Amazon S3. En utilisant le document SSM `AWS-RunRemoteScript` prédéfini ou le `aws:downloadContent` plugin dans un document SSM personnalisé, vous pouvez exécuter des Ansible Playbooks et des scripts en Python, Ruby ou PowerShell, pour n'en nommer 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 plus d'informations, consultez Exécution de scripts depuis GitHub et Exécution de scripts à partir d'Amazon S3. 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 vous permettent de créer et de partager un ensemble standard de documents SSM secondaires Comptes AWS pour des tâches courantes telles que le démarrage d'un logiciel antivirus ou la création d'instances joignant un domaine. Vous pouvez exécuter des documents composites et secondaires stockés dans Systems Manager ou Amazon S3. GitHub Après avoir créé un document composite, vous pouvez l'exécuter en utilisant le document SSM prédéfini `AWS-RunDocument` . Pour plus d'informations, consultez Création de documents composites et Exécution de documents à partir d'emplacements distants.	26 octobre 2017

Modification	Description	Date de publication
	Référence plug-in de document SSM : pour faciliter l'accès, nous avons supprimé la référence plug-in SSM pour les documents SSM de la référence d'API Systems Manager pour l'intégrer au Guide de l'utilisateur. Pour plus d'informations, consultez Référence de plug-in de document Command.	
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 consulter 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 Utilisation des versions de paramètre.	24 octobre 2017
Prise en charge du balisage des documents Systems Manager	Vous pouvez désormais utiliser l'AddTagsToResourceAPI, le ou le AWS CLI AWS Tools for PowerShell pour étiqueter les documents de Systems Manager 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 Balisage des documents Systems Manager.	3 octobre 2017

Modification	Description	Date de publication
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 Utilisation de Systems Manager dans des environnements hybrides et multicloud avec des informations pour Raspbian Linux. • Mise à jour Utilisation de Systems Manager avec des instances EC2 avec de nouvelles exigences relatives aux Windows Server instances. SSM Agentnécessite Windows PowerShell 3.0 ou version ultérieure pour exécuter certains documents SSM sur des Windows Server instances (par exemple, l'ancien document `AWS-ApplyPatchBaseline` SSM). Vérifiez que vos instances Windows Server exécutent Windows Management Framework 3.0 ou version ultérieure. Le cadre inclut PowerShell. Pour de plus amples informations, consultez Windows Management Framework 3.0.	2 octobre 2017
Résoudre le problème des instances Windows inaccessibles à l'aide du flux de travail Automation EC2Rescue	EC2Rescue peut vous aider à diagnostiquer et à résoudre les problèmes qui peuvent survenir sur les instances Amazon EC2 Windows Server. Vous pouvez exécuter l'outil en tant que flux de travail d'automatisation de Systems Manager en utilisant le document AWSSupport-ExecuteEC2Rescue. Le document AWSSupport-Execute EC2Rescue est conçu pour exécuter une combinaison d'actions de Systems Manager AWS CloudFormation , d'actions et de fonctions Lambda qui automatisent les étapes normalement requises pour utiliser EC2Rescue . Pour plus d'informations, consultez Exécuter l'outil EC2Rescue sur les instances inaccessibles.	29 septembre 2017
SSM Agent installé par défaut sur Amazon Linux	Par défaut, SSM Agent est installé sur des AMIs basées sur Amazon Linux et datées de septembre 2017 et après. Installez manuellement l'SSM Agent sur les autres versions de Linux, comme décrit dans Utilisation de SSM Agent sur des instances EC2 pour Linux.	27 septembre 2017

Modification	Description	Date de publication
Améliorations de Run Command	Run Command inclut les améliorations suivantes. • Vous pouvez restreindre l'exécution de commandes à des instances spécifiques en créant une politique 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 Restriction de l'accès Run Command en fonction des balises. • 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 Exécuter des commandes à grande échelle.	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 SSM Agent les journaux vers Amazon CloudWatch Logs	Vous pouvez désormais apporter une simple modification de configuration à vos instances pour y SSM Agent envoyer des fichiers journaux CloudWatch. Pour plus d'informations, consultez Envoi de journaux SSM Agent à CloudWatch Logs.	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 Démonstration : utiliser la synchronisation de données de ressources pour regrouper les données d'inventaire.	1 septembre 2017

Modification	Description	Date de publication
Nouvelles procédures State Manager	Ajout de deux nouvelles procédures à la documentation State Manager : Démonstration : Mise à jour automatique de l'SSM Agent (CLI) Procédure : Mettre à jour automatiquement les pilotes PV sur les instances EC2 pour Windows Server (console)	31 août 2017
Conformité de la configuration Systems Manager	Utilisez la conformité de configuration pour analyser votre parc d'instances gérées afin de rechercher des incohéren ces de conformité et de configuration de correctifs. Vous pouvez collecter et agréger des données provenant de plusieurs Comptes AWS sources Régions AWS, puis explorer des ressources spécifiques non conformes. Par défaut, le service Configuration Compliance affiche les données de conformité relatives aux correctifs Patch Manager et aux associations State Manager. 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 Conformité d'AWS Systems Manager.	28 août 2017
Nouvelle action Automation : `aws:execu teAutomation`	Exécute un flux de travail Automation secondaire en appelant un runbook Automation secondaire. Avec cette action, vous pouvez créer des runbooks 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 runbooks Automation en supprimant la nécessité de dupliquer les étapes sur les runbooks similaires. Pour plus d'informations, consultez `aws:executeAutomation` - Exécuter une autre automatisation.	22 août 2017

Modification	Description	Date de publication
L'automatisation comme cible d'un CloudWatch événement	Vous pouvez démarrer un flux de travail d'automatisation en spécifiant un runbook d'automatisation comme cible d'un CloudWatch événement Amazon. Vous pouvez démarrer des flux de travail selon un calendrier ou lorsqu'un événement AWS système spécifique se produit. Pour plus d'informations, consultez Exécution d'automatisations basées sur les événements.	le 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 un quota 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 AWS Systems Manager State Manager.	le 21 août 2017

Modification	Description	Date de publication
Modifications apportées à Maintenance Windows	Maintenance Windows inclut les modifications et améliorations suivantes : • Auparavant, Maintenance Windows ne pouvait effectuer de tâches qu'en utilisant Run Command. Vous pouvez désormais effectuer des tâches à l'aide de Systems Manager Automation AWS Lambda, et 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 Run Command et Automation. • Tous les Run Command paramètres sont désormais pris en charge DocumentHash, y compris DocumentHashType TimeoutSeconds,,, Comment et 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 AWS Systems Manager Maintenance Windows.	le 16 août 2017

Modification	Description	Date de publication
Nouvelle action Automation : `aws:approve`	Cette nouvelle action des runbooks Automation interrompt temporairement l'exécution d'Automation jusqu'à ce que les principaux 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 Référence sur les actions Systems Manager Automation.	le 10 août 2017
L'automatisation assume un rôle qui n'est plus nécessaire	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 Automation : • Lorsque vous souhaitez restreindre les autorisations d'un utilisateur sur une ressource tout en souhaitant que l'utilisateur puisse exécuter un flux de travail Automation nécessitant des autorisations supérieurs. Dans ce scénario, vous pouvez créer un rôle de service avec des autorisations supérieures 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 Configuration d'Automation.	3 août 2017

Modification	Description	Date de publication
Conformité de la configuration	Utilisez la conformité de configuration Amazon EC2 Systems Manager pour analyser votre flotte d'instanc es gérées afin de rechercher des incohérences de conformité et de configuration de correctifs. Vous pouvez collecter et agréger des données provenant de plusieurs Comptes AWS sources Régions AWS, puis explorer des ressources spécifiques non conformes. Pour plus d'informa tions, consultez Conformité d'AWS Systems Manager.	8 août 2017
Améliorations de document SSM	Les documents de commande et de politique 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. Les documents SSM Command utilisant la version de schéma 2.0 ou ultérieure peuvent désormais inclure plusieurs plugins du même type. Par exemple, vous pouvez créer un document Command qui appelle plugin `aws:runRunShellScript` plusieurs fois. Pour plus d'informations sur les modifications apportées à la version 2.2 du schéma, consultez les documents AWS Systems Manager. Pour plus d'informations sur les plug-ins SSM, veuillez consulter la rubrique Référence du plug-in de document Commande.	12 juillet 2017

Modification	Description	Date de publication
Application des correctifs Linux	Patch Manager peut désormais corriger les distributions Linux suivantes : Systèmes 64 bits et 32 bits • Amazon Linux 2014.03, 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 Linux 2015.03, 2015.09, ou version ultérieure • Red Hat Enterprise Linux (RHEL) 7.x ou version ultérieure Pour plus d'informations, consultez AWS Systems Manager Patch Manager. **Note** • Pour corriger les instances Linux, vos instances doivent exécuter SSM Agent version 2.0.834.0 ou ultérieure. Pour plus d'informations sur l'agent, consultez la section intitulée Exemple : Mise à jour de l'SSM Agent dans Exécution des commande à partir de la console. • Le document SSM `AWS-ApplyPatchBaseline` est remplacé par le document `AWS-RunPatchBaseline` .	6 juillet 2017

Modification	Description	Date de publication
Synchronisation de données de ressources	Vous pouvez utiliser la synchronisation de données de ressources Systems Manager pour envoyer les données d'inventaire collectées à partir de 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. Toutes les données d'inventaire étant stockées dans un compartiment S3 cible, vous pouvez utiliser des services tels qu'Amazon Athena et Amazon QuickSight pour interroger et analyser les données agrégées. Pour plus d'informations, consultez . Configuration de la synchronisation de données de ressource pour Inventory Pour obtenir un exemple d'utilisation de la synchronisation des données de ressource , consultez Démonstration : utiliser la synchronisation de données de ressources pour regrouper les données d'inventaire.	29 juin 2017

Modification	Description	Date de publication
Hiérarchies de 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 Utiliser des hiérarchies de paramètres. Pour obtenir un exemple d'utilisation des hiérarchies des paramètres, consultez Utiliser des hiérarchies de paramètres.	22 juin 2017
Prise en charge SSM Agent pour SUSE Linux Enterprise Server	Vous pouvez installer SSM Agent sur SUSE Linux Enterprise Server (SLES) 64 bits. Pour plus d'informations, consultez Utilisation de SSM Agent sur des instances EC2 pour Linux.	14 juin 2017

Conventions de rédaction

Les conventions typographiques courantes pour le Guide de l'utilisateur AWS Systems Manager sont les suivantes.

Exemples différenciés pour des systèmes d'exploitation locaux ou les langages de ligne de commande

Nous utilisons les onglets pour présenter les commandes en fonction du type de système d'exploitation local d'un utilisateur. Pour Linux et macOS, nous utilisons la barre oblique inverse (\) pour diviser les longues commandes en plusieurs lignes. Pour Windows Server, nous utilisons le caret (^) pour diviser les commandes en plusieurs lignes.

Exemple :

Linux & macOS

```
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
```

Éléments de l'interface utilisateur

Mise en forme : texte en gras

Exemple : Choisissez Fichier, Propriétés.

Entrée utilisateur (texte qu'un utilisateur tape)

Mise en forme : texte dans une police monospace

Exemple : Pour le nom, saisissez **`my-new-resource`**.

Texte d'espace réservé pour une valeur requise

Mise en forme : texte en *italique*

Exemple :

```
aws ec2 register-image --image-location DOC-EXAMPLE-BUCKET/image.manifest.xml
```

AWS Glossaire

Pour la AWS terminologie la plus récente, consultez le AWS glossaire dans la Glossaire AWS référence.

Les traductions sont fournies par des outils de traduction automatique. En cas de conflit entre le contenu d'une traduction et celui de la version originale en anglais, la version anglaise prévaudra.